

DEPARTMENT OF INFORMATICS

TECHNISCHE UNIVERSITÄT MÜNCHEN

Bachelor's Thesis in Informatics: Games Engineering

**Learning Japanese Kanji Using Heisig  
Method Through AR And Deep Learning**

Zixiang Wang

DEPARTMENT OF INFORMATICS

TECHNISCHE UNIVERSITÄT MÜNCHEN

Bachelor's Thesis in Informatics: Games Engineering

**Learning Japanese Kanji Using Heisig  
Method Through AR And Deep Learning**

**Japanische Kanji Lernen Mit Heisig  
Methode Durch AR Und Deep Learning**

Author: Zixiang Wang  
Supervisor: Prof. Gudrun Klinker, Ph.D.  
Advisor: Eichhorn, Christian  
Submission Date: November 16

I confirm that this bachelor's thesis in informatics: games engineering is my own work and I have documented all sources and material used.

Munich, November 16

Zixiang Wang

## Acknowledgments

I would like to show my great gratitude to my advisor Christian Eichhorn for his continued support and patience throughout this project.

Secondly, I hope to say thank you to Lei Mao for publishing and sharing the model freeze code on GitHub so that I could find the solution to the interface between C and Python.

Finally, I wish to thank my family for their constant support and encouragement.

# Contents

<b>Acknowledgments</b>	<b>iii</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Motivation . . . . .	1
1.2 Dragon Tale – A Serious Game for learning Japanese Kanji . . . . .	2
<b>2 Related Work</b>	<b>3</b>
2.1 Serious game . . . . .	3
2.1.1 Advantage . . . . .	3
2.1.2 G/P/S model . . . . .	4
2.1.3 Casual game . . . . .	5
2.1.4 Game flow . . . . .	5
2.1.5 DDA: Dynamic Difficulty Adjustment . . . . .	6
2.2 Japanese Kanji System . . . . .	8
2.2.1 Source . . . . .	8
2.2.2 characteristic: pictographic . . . . .	8
2.2.3 characteristic: radical . . . . .	9
2.2.4 characteristic: ideograph . . . . .	9
2.2.5 characteristic: phonogram . . . . .	10
2.3 Edutainment related to Japanese kanji learning game . . . . .	10
2.3.1 Repeated memory method . . . . .	10
2.3.2 Effect of humour . . . . .	11
2.3.3 Reasonable learning curve . . . . .	12
2.3.4 Feedback . . . . .	13
2.4 Core theoretical point related to implementation . . . . .	14
2.4.1 Edgar Dale’s theory . . . . .	14
2.4.2 Heisig method . . . . .	16
<b>3 Software Requirement</b>	<b>19</b>
3.1 Unity3D . . . . .	19
3.2 Vuforia Library . . . . .	19
3.2.1 What is AR . . . . .	20
3.2.2 Vuforia Set-up . . . . .	21

## Contents

---

3.2.3	Simplest AR example . . . . .	21
3.2.4	OnTrackingfound and OnTrackinglost . . . . .	22
3.3	Jupyter Notebook . . . . .	23
3.4	TensorFlow . . . . .	24
3.4.1	Definition of Deep learning . . . . .	24
3.4.2	Google Colab . . . . .	25
<b>4</b>	<b>Approach</b>	<b>26</b>
4.1	Image comparison using TensorFlow . . . . .	26
4.1.1	Deep learning model . . . . .	27
4.1.2	Overfitting and underfitting . . . . .	28
4.1.3	Data augmentation . . . . .	29
4.1.4	Freeze the model . . . . .	30
4.1.5	Encountered problem . . . . .	30
4.1.6	Prediction result . . . . .	30
4.2	Load game data using XML file . . . . .	31
4.2.1	General structure of XML . . . . .	31
4.2.2	code of XML part . . . . .	32
4.3	Apply deep learning model in Unity using TensorFlowSharp . . . . .	33
4.3.1	TensorFlowSharp . . . . .	33
4.3.2	Code of TensorFlowSharp . . . . .	33
4.4	Painting function in Unity using LineRenderer . . . . .	34
4.5	Animation Design using extended Heisig Method . . . . .	35
4.5.1	instruction book method . . . . .	35
4.5.2	story method . . . . .	36
4.5.3	Correlation method . . . . .	37
<b>5</b>	<b>Frame work</b>	<b>39</b>
5.1	Game Design Document . . . . .	39
5.1.1	Basic Information . . . . .	39
5.1.2	Game Story . . . . .	39
5.1.3	Level Design . . . . .	40
5.2	Project Frame Work . . . . .	40
5.2.1	Start Scene . . . . .	40
5.2.2	Main Scene and Tech Tree . . . . .	41
5.2.3	Marker Scene . . . . .	42
5.2.4	Quests panel . . . . .	42
5.2.5	Drawing game . . . . .	42
5.2.6	Feedback . . . . .	44

## *Contents*

---

5.3 Game Flow Analyse . . . . .	44
<b>6 Evaluation</b>	<b>46</b>
6.1 Evaluation before development . . . . .	46
6.1.1 Target group analyse and Personas . . . . .	46
6.2 Expected results . . . . .	47
6.3 Evaluation after development . . . . .	47
6.3.1 Participants . . . . .	47
6.3.2 Questionnaire and System Usability Scale . . . . .	48
6.3.3 Evaluation result from Questionnaire . . . . .	48
<b>7 Future work</b>	<b>51</b>
7.1 Gameplay improvement . . . . .	51
7.1.1 Game theme and game genre determining . . . . .	51
7.1.2 Game storyline and game world view . . . . .	52
7.2 Resource improvement . . . . .	53
7.2.1 Animation and 3D model . . . . .	53
7.2.2 Training model . . . . .	53
<b>8 Conclusion</b>	<b>55</b>
<b>List of Figures</b>	<b>56</b>
<b>List of Tables</b>	<b>58</b>
<b>Bibliography</b>	<b>59</b>

# 1 Introduction

## 1.1 Motivation

With the passage of time, however, the world isn't limited by one single culture. People show great interest in other cultures which seem totally indifferent with the culture they living in. As a bridge linking cultures, foreign languages play such a role. Starting from elementary school, the second language education is of equal importance as other basic subject. It opens the window towards other unknown but novel cultures. Especially does Japanese culture disseminate by its famous anime, Ninja culture and Bushido spirit with great charm. However, some problems occur in the non-Asian countries when the people there try to learn Japanese. Because of the differences between Western and Oriental language systems, Kanji Symbol causes huge difficulty owning to its diversity and complexity. So it is necessary for the game designers researching in this direction.

Recent years more and more serious games with Japanese theme appear on the market. And there are also many good works among them. 'Learn Japanese to Servile' is a brilliant series vended on steam for already four years. Relying on substantial sales and good reputation this game publisher published 'Learn Japanese to Servile' series game once a year, first was hiragana game and then came katakana game into public sight, at last Kanji game. They stand out not only for their appealing story line but also a good flow control of the players which can induce player playing game till the end. This is a good example as a commercial video serious game. But on the contrary there also exist other games which failed on market. After analysing the reasons behind by myself, I Suspect the problems that these games lose the control of game flow, and the challenge of kanji remembering ruins the game experience. In other words it is improperly difficult for the player mechanically remembering, learning and understanding the kanji symbol which isn't in their first language systems. So I wrote this thesis for putting my opinion on the table and providing a potentially feasible solution helping with the serious game learning Japanese Kanji.

## 1.2 Dragon Tale – A Serious Game for learning Japanese Kanji

Dragon Tale is a 3D serious game teaching player Japanese kanji by using Augmented reality. With a rich storyline the game is about an adventure protecting a baby dragon and exploring the ancient language "Kanji". This game absorbs the advantages of other serious game about teaching Japanese and avoids the shortcomings. For example compared to the game "Learn Japanese to Survive!" series which I think a fine commercial game winning good comments on steam, Dragon Tale obtains the AR elements and a mythic 3D world while the other doesn't.

The basic game mechanics of Dragon Tale are a little bit complicated as there are several mini games in charge of different knowledge points, that ensures the game flow and encourages the learning process. Mini game 1: To help player memorize the right kanji stroke order, a drawing game is designed to achieves the goal with repetitive training method. When player writes incorrectly, the hint will be given in the form of animation showing the right stroke order.

Mini game 2: To teach player meaning of each kanji, there is a quiz game requiring player to select the right kanji from six based on the given image which matches the meaning of it.

Mini game 3: A reading game focuses on the respect of pronunciation of each kanji where player should drag the right hiragana out of a pool into a predefined slot to defeat the samurai.

Mini game 4: The fourth mini game is about AR. In the normal cases when player shows a single marker front of the augmented camera, the corresponding object will show up. But in this mini game when player places two markers together in a right order to build an advance words, the combined word's meaning will appear as a new virtual object on the screen.

Mini game 5: When encountering with enemies, it will enter a turn-based game allowing player to draw kanji so that the kanji will transform to corresponding attack by dragon.

## 2 Related Work

### 2.1 Serious game

Serious game is a game with the primary purpose other than pure entertainment. Its purpose is so serious that may aim at a realistic goal. In some ways for example it can be used to train a specific physical ability which involves mostly in the sports domain. Other purposes such as health care, scientific exploration or politics are still able to integrate into serious games. And above all, it has also a place in the language teaching domain which will be discussed in this thesis of great length.

#### 2.1.1 Advantage

So why serious games instead of traditional teaching systems? The answer is that with active engagement, serious games lead to discovery, observation, trial and error and problem solving, important aspects of learning [Dic05]. Figure 2.1 below shows a flow chart explaining how serious games transform a knowledge point to a long-term memory. The Game distracts the attention of players from mechanical learning and appeals the attention by rewarding or accomplishment. Research shows that a positive atmosphere increases student academic achievements and leads to increased self-esteem, which yields twice the result with half the effort. And that's why teachers often try to arrange mini games to assist in teaching plans and learning process.

Games can make learning fun (Westera, 2015).

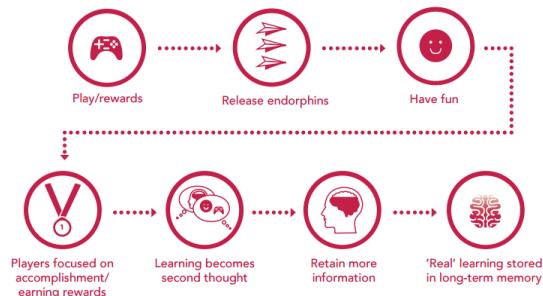


Figure 2.1: How do serious games help remembering (from [Pea16])

Secondly serious games installed on IPad, Smartphone,or other platforms realizes mobile learning process, which means you could get into study state in the fractional free time, whenever for example sitting on the bus home, or even waiting for the dinner at the table.

The third point of serious games' advantage compared to traditional learning is that this kind of informal learning takes place primarily outside school and is a means of acquiring knowledge with an interesting game as a motivation factors to begin and persist in. For some students weary of study, language learning are regarded as a goal of learning which belongs to a sort of burden for them. But what if language learning being a part of tool for gaming? The latter will absolutely become inspiration and motivation for those 'naughty' students.

### 2.1.2 G/P/S model

G/P/S model is a classification system for all the video games.[DAJ11] The intention of this classification for our games designer is to make clear the target player, designed purpose and game structure so that the product can exactly match customer expectations. After decomposing the model, three abbreviations concepts turn up. Their meanings are as follow:

G for Game-play: which refers to the game structure or the game-play type. Summarizing the concept in one sentence is: How to play this game. Generally speaking there are two broad headings of video game, one is so called Game-based with a clearly defined game goal, the other is so called Play-based lacking game goals.

P for Purpose: which refers to the design purpose. As for most of the normal video games, it is entertainment .As for language learning serious games, it is training

S for scope: which means targeted application of the game. Clear said the target market and the target user. For instance the 'The Tactical Language and Culture Training System' is designed mostly for the US military,whose users are in the army.The game 'Mondly' serves the general public facing to all the people in various age groups

Classifying the game with this model will help both users and designers getting a first impression of the product. How should the designers design for this target group? Which kind of game play rules are to defined? Or for the users :Does this game meet my original demand? Does it fit in me?

### 2.1.3 Casual game

Casual games opposed to serious games have basically simple rules, short gameplay duration and less required game abilities for playing.[Gro19] The characteristics of casual games are listed in formal definition below which came from :

- 1.Instant play which means easy to learn the game rule.
  - 2.Quick play which means player could get pleasure within a short time.
  - 3.Common play which means the game addresses a vast majority of player types.[Juu10]
- Casual games are likely to choose puzzle solving, simplistic graphics, and straightforward play mechanics, which are all the concepts of "Mini game". Mini games based gameplay involves multiple kinds of play mechanics and never boring the players since the freshness of attempting new mini games tempts players' desire to continue. It can be regarded and used as a motivation of language learning which meets our design goal. Even more, the active atmosphere of casual game can neutralize the difficulty in the language learning process for those people who go "numb" facing the foreign languages. In game flow theory, it pulls the flow line back to the diagonal middle zone from anxiety zone caused by challenge.

### 2.1.4 Game flow

20 years ago, with an intention to explain happiness, Mihaly Csikszentmihalyi found Flow, the feeling of complete and energized focus in an activity, with a high level of enjoyment and fulfillment.[CB10] Csikszentmihalyi developed this theory to find a way guiding player to a better game experience. One famous model was invented named flow zone which is a two-dimension graph with challenge as vertical axis and abilities/skill as horizontal axis. The area near the angular bisector is called "flow" which means a good game experience for player. Located in the bottom of the "flow" area there is "bored" area and in the above is the "anxiety" area. This model explain that the player's abilities and game challenge must be positive correlation, or too hard may cause anxiety and too easy may cause boredom.

The flow zone can be expanded as well. An appropriately difficult game is called hardcore game which goes after intensive player immersion. An appropriately easy game is called casual game or the designed for novices. Under ordinary circumstance however, hardcore games are typified by their steep learning curve, cutting edge graphics, high production cost, and long gameplay duration [Con11]. As for a game aiming at teaching language the hardcore definition tends to the steep learning curve, tight learning schedule. To visualize this, think of a learning plan that requires the

learner to get a good command of spoken English as soon as they learned ABC yesterday. The example above is a little bit exaggerated but exactly demonstrates anxiety out of the flow zone. Serious game as a kind of informal learning is supposed to provide with a new learning approach in the off school time. It can be serious but please never as an another burden or anxiety for the learners. The expected effect that such serious games pursue is a transmitting study in a somewhat pleasant way that evokes interest and positive emotions.[Gie15] So therefore normal games in the middle flow zone and casual games is a good choice for serious games.

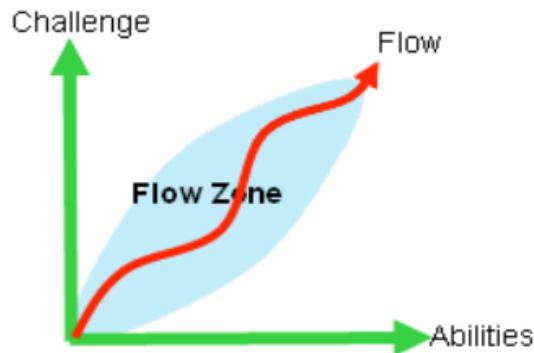


Figure 2.2: Player encounters positive emotion in the game flow zone (from [Che07])

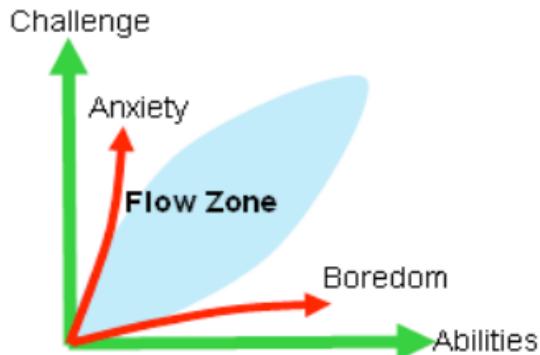


Figure 2.3: Player encounters negative emotion out of the game flow zone (from [Che07])

### 2.1.5 DDA: Dynamic Difficulty Adjustment

Some readers might still think as they get here that it isn't of much use even if learning in a casual way. The only thing becoming better is emotion but this helps little. So

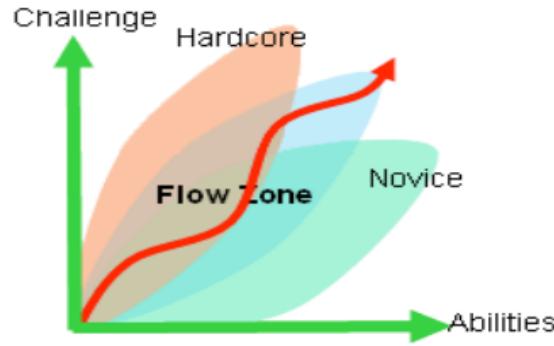


Figure 2.4: The expanded game flow zone (from [Che07])

based on the huge gap of language talent between different players, the Dynamic Difficulty Adjustment theory is created for better game adaptation. This theory emphasizes on providing players with subconscious choices to help them actively customize their optimal video games experiences rather than imposing restrictions on the only difficult level. Following this rule, most of the video games on the market ask player to choose the degree of difficulty at the beginning, such as begin as a novice, old hand or professional. The difficulty can be changed not only in the beginning settings, it is also possible to be controlled in the game by players own. A typical means of that is setting multiple missions at the same time, whose degrees of difficulty differ from each other. In that way player has the autonomy to adjust its emotion to pleasure. This sense of control enables the navigation of flow experience, which is shown as figure 2.5 below.

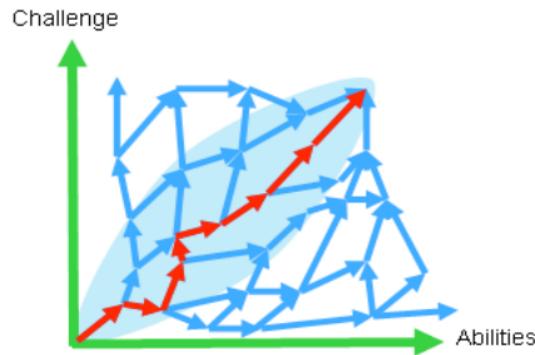


Figure 2.5: Game flow adjustment through choices (from [Che07])

## 2.2 Japanese Kanji System

### 2.2.1 Source

Japanese kanji was first brought into Japan by letters, seals, mirrors, or Swords from China[KOI10]. As Kanji was introduced in the ancient time when Japanese people got used to write Hiragana and Katakana, they only assimilate the way how kanji is written, but the pronunciation of each kanji was defined by themselves in their own language system. As time goes by, Chinese Kanji has been simplified from traditional Chinese to simplified Chinese, but this Kanji revolution didn't influence the kanji in Japan. So today we could find out that the kanji between these two countries with the same meaning are a little bit difference in writing way.

Trace back to the ancient times when kanji was first 'painted' on the sell to record something. It is described with "painted" because the formal kanji system wasn't defined then. Asian used graphic patterns to transform information to partners. Imagine that, once an Asian primitive man looked upon at the sky and saw yellow shining "ball" hanging there which is named the "sun" today. He recorded the shining "ball" by marking on the tree. However when night came and the moon appeared, he was confused that this white shining "ball" was similar to the ball morning but looked indifferent. He recorded this one again next to the "sun" mark with a similar but distinguishing graphic.[Jia07] In the years that followed those patterns evolved into kanji with this pictographic characteristic.

### 2.2.2 characteristic: pictographic

Pictograph is one characteristic of kanji, which brings convenience for language learners recognizing what the kanji means. Like the two examples above, "日"(the sun) and "月"(the moon) are typical enough to describe how pictograph works. Look at the sun kanji, the outer rectangle is the shape of the sun and the middle horizontal line represent the shining light. As for the moon it isn't an outer rectangle because sometimes moon is curved for astronomical and geographical reasons, and those two horizontal strokes can be imagined as clouds in front of the moon.

An another example is given between "木"(wood), "林"(woods)and "森"(Forest). Look at these three kanji we could figure out that the latter kanjis are repeated stack of the first "wood". Beyond doubt, the more "木" there is, the more woods the kanji means. This means of expression offers intuitive sense for the Japanese beginners.

### 2.2.3 characteristic: radical

Most Kanji are classified in kanji dictionaries according to their main components which are called radicals (roots) in English and 部首 (ぶしゅ) in Japanese. 部 (ぶ) means a group and 首 (しゅ) means a head. Radicals are transformed from some simple kanji and retain the original meaning. The transformation of single kanji to radical is resizing, rescaling or simplifying and at last the radical stays always at a fix position which can be right aside, left aside, bottom aside, top aside or outer surrounding. If you encounter with a kanji but don't understand the whole meaning of it. Don't worry cause it is possible to decompose this kanji into radical and the rest parts. It is same as some words English system, such as fireman, firework, fireplace. The word beginning with fire definitely has a connection between fire. Take a look at the concrete examples below : 手(hand) is simplified to 扌 and this radical stays always left : 持(hold) 扌(strike) 摘(pick), which means do something by hand.

刀(knife) is simplified and the radical stays always right: 刂(carve) 削(pare), which means do something with knife.

口(mouth) is resizing and the radical stays 唱(sing) 呼(breathe out) 吸(breathe in), which means the actions only feasible by mouth[ali13]

### 2.2.4 characteristic: ideograph

The concept here doesn't mean that several kanji compose a word, but explain the origin of a single kanji. Some simple kanji may come from the pictographic but some others don't. The kanji isn't composed of other different single kanjis for no reason, it certainly exists connection between the entirety and parts. Normally, the connection isn't clear at first glance. To figure out it, we need to Incorporate life experience, as language was invented among the daily life. Some Examples here are listed below:[Bee06]

忍(bear/tolerant/Ninjutsu) : this kanji is composed of 刂(blade) and 忄(heart). According to Japanese ninja culture, Ninjutsu is a technique used in assassinate and spy activity. The mission performer is rigorously screened and has already studied Ninjutsu for a so long time that must guarantee success of the mission. Spies or assassins need to hold their horses to catch an opportunity for acting. So one necessary psychological quality for them is tolerant enough. And the connection between blade, heart and tolerant is that Ninja must be tolerant and ignore everything but only focus on the mission even if a blade hanging above its heart.

明(bright): this kanji is easier to understand than the previous one. It is composed of 日(sun) and 月(moon) which bring the light no matter day and night. The kanji was

invented a long time ago, so there existed no lamp or electric torch. This is why the two biggest luminaries on the sky represent brightness.

This rule usually has intersection with radical rule, and the two rules can explain kanji meaning together.

泪(tear) is a kanji with water radical and 目(eye). The liquid secretion from eyes is tear, that is easy puzzle.

### 2.2.5 characteristic: phonogram

Phonogram kanji means one part of the kanji in charge of pronunciation and other parts in charge of the meaning. These Kanji are by far the largest category of ninety percents and almost all the compound kanji follows this rule. One example can be the kanji 銅 which means copper in English and its reads "doo". The left part 金 indicates its meaning and this kanji is metal and the right part indicates its sound.[Bee06]

## 2.3 Edutainment related to Japanese kanji learning game

### 2.3.1 Repeated memory method

As we all studied at school in childhood, we know that the kernel learning mechanic revision. All the things in memory will fade away by time, the more times person recall the thing, the longer it lasts and slower it is forgotten. Only reviewing the knowledge repeatedly can generate long-term memory. One famous German psychologist named Hermann Ebbinghaus made great contribute on memory theory. He presented a view on forgetting curve and claimed humans tend to halve their memory of newly learned knowledge in a matter of days or weeks unless they consciously review the learned material.[Wik20b]

And then the key problem come up: how do serious games require player to review the learned material properly. Indeed, needless to say viewing the learned material, it can't be guarantied whether the game will be played everyday. In order to solve this problem, two methods are recommend as follow:

1: Setting daily gift or daily mission

As a kind of bait to attract player to play the game everyday, daily gift can be a third motivation where the first motivation is the passion to learn and the second motiva-

tion is the desire to play. In that case, under the precondition that player loses the first two motivation. daily gift is the last defence preventing from being uninstalling. It is a key feature to keep freshness and remind player reviewing. As a further idea however, daily gift isn't gained in vain and can be designed as a reward after solving a kanji puzzle or gained after viewing an interesting Japanese kanji history, or even a joke. Hearthstone and MTG area do well in this field, in the interval of update daily mission is the only reason for those people who feel bored and almost give up the game.

2: Setting a reminder:

The reminder works as a supervisor reminding the "forgetful" users of the forgotten software application. Even if the application hasn't been opened on the background of the mobile phone, reminder will send a message trying to urge users to follow the schedule. For learning is a long-time process which requires repeated reviewing, building a daily habit of learning is a successful educational beginning which can last the learning process till the end. Reminder can arrange the gameplay into daily schedule in form at least. Although it perhaps doesn't work well for the most self-willed users because it can't come into play with strong and forceful power like parents or teachers, but in the educational field half a loaf is better than no bread.

### 2.3.2 Effect of humour

There are several types of fun, besides of the fun coming from exploration experience in an unknown stimulation virtual world and the fun accompanied by achievement such as good grades, sense of accomplishment after bringing an issue to a close, a sort of original fun made by joke or humour is worth a place in entertainment. Sticking on one thing for a long time and continuously spending too much time in it may cause dull brain. You could also take a deep breath outside or change to other thing actively, but a passive break is remarkable in game designing. Research shows that learning feeling ease will lowers the psychological filter and is more likely to transform the input in intake which is correct retention in the long-term memory. However excessive humour has its downside that is it has tensity to monopolize the attention, being associated with frivolousness and keep the player in flow[Lom12]. So the best design for humour is use it sparingly and scattered here and there in the serious games.

### 2.3.3 Reasonable learning curve

This part is of much importance that needs to be paid sufficient attention on. As the thesis has already argued above, serious games with steep learning curve will be classified to hardcore game, which may not so suitable for keeping flow. Because essentially foreign language learning has already contained much difficulties which isn't comparable with being cleared by the customs in other hardcore games like Darksoul 3. Further speaking you can pass the game in a week but never learn the language well in the same amount of time. Worse more under the premise of normal human beings if the study requirement or the learning curve mentioned above is ridiculous hard, players will have a bad experience or even uninstall the game. Human beings is the kind of creature which is looking for awards and accomplishment from others and this belongs to a need level in Maslow's hierarchy of needs[Wik20c]. Overdifficult problems coming one after another without encouragement will frustrate the player.

Back to the theme of this thesis, for serious games with Japanese kanji learning goal, designer should avoid complicated kanji at the beginning game stages. For one reason it ruins the game flow and brings a difficult first impression of language learning. For the second reason is that the usage rate of complicated kanjis is lower than simple kanjis', which doesn't accord with the instrumental and practical characteristic of language. Not only avoid complicated kanjis but also words constituted by kanjis. The task load of remembering a word isn't several times equal to a kanji where the times number is the kanji number in the word because the arrange order of the kanjis counts too.

Koe is a 3D RPG game for teaching Japanese common expression , kanjis, words and

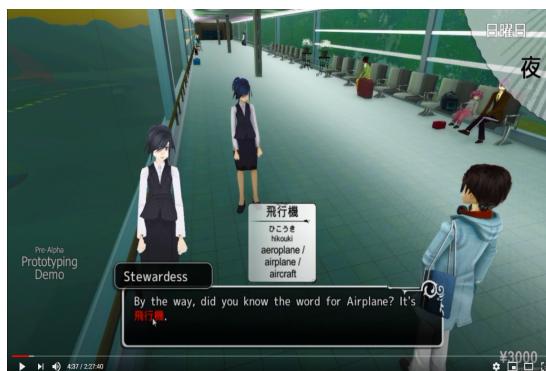


Figure 2.6: A screenshot of Koe(demo) from YouTube teaching complicated word (from[The17])



Figure 2.7: A screenshot of Koe(demo) from YouTube teaching easy expression (from [The17])

their meaning[GAM]. And there is an example I mentioned above in it. Figure 2.6 is a screenshot cut by me on YouTube showing at the beginning of the game or exactly when the subscriber played the game four and a half minutes the word 飛行機 appears. And the figure 2.7 shows the game teaching はじめまして .It is a Japanese greeting and the commonest expression which means "Nice to meet you at the first time". These two examples conform to a steep learning curve and is meaningless appearing simultaneously. For the novice はじめまして is always teached at the first lesson. But the word 飛行機 is a complicated word in high level. As a suggestion to assist player learning kanji joyfully, kanjis like numbers from one to ten with less than five strokes are recommended for the beginning phase of game. Here I also advise not to require the pronunciation or polyseme of kanjis in a single kanji learning game. The outcome of the game should be just recognizing the kanji when it appears somewhere. It will be an overload for player learn the kanji writing, pronunciation and the meaning at the same time. In my opinion pronunciation learning is the task of katakana and hiragana game, and the both phases are supposed to involve the meaning learning. Relying on various of Kanji learning mechanic including heisig's method, the connection between meaning and Kanji writing is easy to establish. There isn't a given theory between how to write and how to pronounce, so one kanji learned twice at different perspectives is necessary and the meaning will act as a bridge connecting and strengthening the memory.

### 2.3.4 Feedback

Immediate feedback about the progress that is being made is an indispensable condition of game flow. The requirement of an immediate feedback doesn't only include the

system response time after clicking, it also means a kind of player-game interaction. For some young player who is self-doubt, gaining recognition after success or getting encouragement after failure is the psychological support measures that game could offer in compensation for lack of care in players' real life.

Another feedback serves the quality of education. The intelligent tutoring system (ITS) is a computer program or computing device that factors student performance into when and how it generates and provides guidance[HLA13]. ITS can be decomposed into four models which are:

1. Expert model, it represents subject matter expertise and provides the ITS with knowledge of what it's teaching.
2. Student model, it represents what the user does and doesn't know, and what he or she does and doesn't have.
3. Instructor model, it accepts information from expert and student model and enables the ITS to know how to teach, by encoding instructional strategies used via the tutoring system user interface.
4. User interface model, the interface between end user and ITS which reaches a communication goal.

This program aims to let the system replace a human teacher and provides appropriate advice to learner to form one-to-one teaching model. Various of feedback according to performance makes a sense of reality and create an more immersive learning environment compared to simply feedback with single style. Then one question is how does the intelligent tutoring works in the instances and what does it look like?

Imagine that there are two players playing a serious game learning Japanese kanji. In one certain Unit there is a mini game asking for building a compound word by dragging the kanjis in the slots. The correct answer is 日本語(Japanese) but one player builds a 白本語 and the other builds a 日本五. The mistake from previous player is confused 日(sun) with 白(white) because of the written similarity. And on the other hand, the mistake from latter player is confused 語(language) and 五(five) because of the same pronunciation. These two mistakes belong to different perspectives and are supposed to get different feedback. The feedback could be explaining the potential mistake reason or tending the training point to somewhere else.

## 2.4 Core theoretical point related to implementation

### 2.4.1 Edgar Dale's theory

Edgar Dale's work continues to influence educational technologists in the 21st Century and put forward a theory called "Dale's Cone of Experience"(see figure 2.8)[BAA13]. In this pyramid the abstraction experience levels are classified where the top stands for

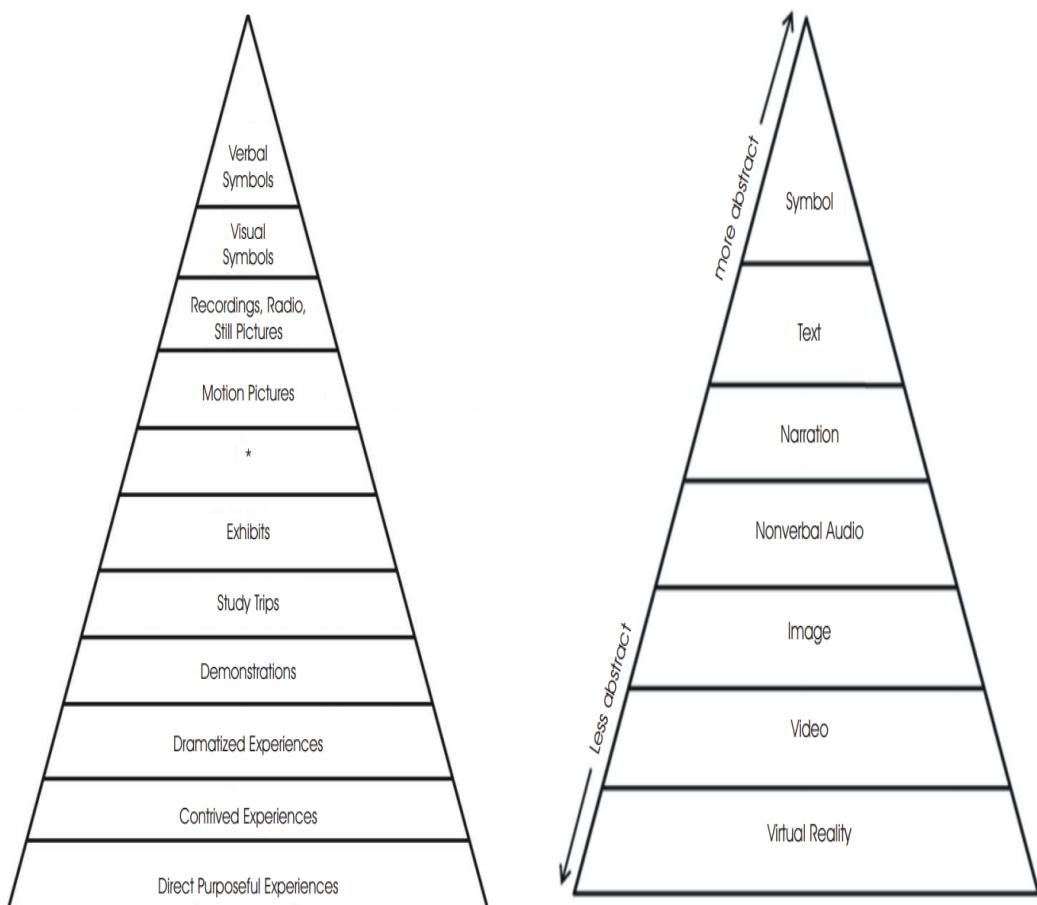


Figure 2.8: Dale's Cone of Experience(from [BAA13])

Figure 2.9: Multimedia Cone of Abstraction  
(from [BAA13])

the most abstract experience and the bottom stands for the most concrete experience. This pyramid doesn't attribute worth of each level such as the educational method is always better than the one below or above. Edgar claims when the learner has no previous experience or foundation with a subject, more direct or more concrete method is needed. But for some other cases, abstract method like symbolic expression may be preferred. For example preparing a theory exam of java programming, a proper way is reviewing the slides or the old examination questions but not programming on computer. The neighbor levels will offer nearly experience and some levels below might present an more abstract effect than the level above in some case. It can be happened because it also partly depends on the learners, studying purpose and the teaching content.

Some years later, Edgar Dale's cone of Experience theory was updated with the development of other areas like memory theory and cognitive load theory. A new Multimedia Cone of Abstraction (MCoA) was put forward whose model is showed in figure 2.9. The proposed MCoA focuses on learning experiences based on their media components and the level of abstraction required of users. More clearly, it emphasises on the technical media components for enhancing learning especially via computer. Take a look at figure 2.8 and 2.9, if we compare the two models horizontally, we could find a common point that the closer a cone to the bottom of the pyramid, the more realistic it is. And the closer a cone to the top of the pyramid, the more abstract it is. However it still doesn't mean that a realistic is better than any other more abstract cone or the reverse meaning. It just offers a clear guidance for the instructional designers who want to use technique in education. And luckily, we ,as serious games designer can still cut the melon from it.

For there exists no level dominates any other and the difference of preferences by learners. A combined learning methods involving both abstract and realistic cones is recommended by me. To play up to all the learners' requirement at a most reachable degree as possible, examples like a kanji with different learning aspect which can be symbol, Image or video will be shown in the latter chapter.

#### 2.4.2 Heisig method

Back to the topic of this thesis, if the target users come from non-Asian areas and almost haven't contacted the east Asian languages ever before, a concrete way to introduce kanji is more accessible. Due to this reason heisig's method was created for introducing the kanji from words aspect to graphic aspect.

James W. Heisig is an author and philosopher who is famed among students of the Japanese and Chinese languages for his Remembering the Kanji and Remembering the Hanzi series. Remembering the Kanji is a series of three volumes, first time published

in 1977 and intended to teach the 3,000 most frequent Kanji to students of the Japanese language. The book offers a new perspective of learning Japanesees kanji by breaking down the complexities of the Japanese writing system into its basic elements and suggesting ways to reconstruct meanings from those elements.[Hei07].The chapters are named as Stories, plots and elements which are the method names used on kanjis from simple to complex.

1.Stories method intends to tell a story about key kanjis and their primitive meanings.  
2.Plots method simplifies a story into short plots,leaving it up to you to patch together the necessary details in a manner similar

3.Element focuses on the ordering of the characters according to their primitives or so-called "element" which belong to a relatively difficult kanji. Combining with the story and plots learnt before, these elements could be worked into a kanji story so as to create a more vivid ambience for the primitive elements to interact.[Hei07]

However, the distinction between these three methods aren't evident since all the kanji explanations from the book are in a text form and whether it is by story method or plot method, the image learner should create is up to themselves, that means some learners lack of ability to understand abstract things, here is the text form explanation words, are of defective benefit on learning with the book. As argued in the previous subsection by the theory of Edgar Dale, the education is better of various forms so that can adjust different kinds of learners' need.

An example that I can hard to get the point is the kanji friend(朋).

In the figure 2.10 says that companion can be remembered as a story about Eve in

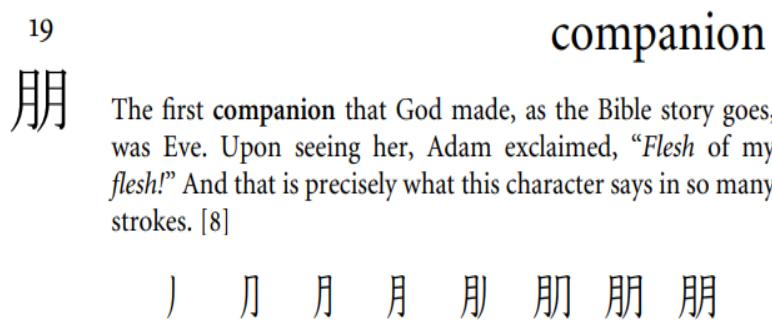


Figure 2.10: How is friend explained in "Remembering the Kanji" (from [Hei07])

Bible. "Flesh of my flesh!" may be a sentence in Bible but as a college student from east, the religion stuff especially Christian culture doesn't influence on me much and barely leave track in my cognitive range, that can't results in understanding the explanation and making a strong memory. Bible is so abstract and unfamiliar to me that I have

## *2 Related Work*

---

not even opened it and read once yet. And according to the article "Why We're More Likely To Remember Content With Images And Video" it proposes and argued about a theory that people remember image or video information more firmly than other types of information including text.[Unk14] So the extended Heisig method that I will introduce in the following chapter will probably more or less help in that respect

# **3 Software Requirement**

In this part I will briefly introduce some software I used in the project development process, which are mainly Unity3D including its extended available function and Jupyter Notebook using TensorFlow.

## **3.1 Unity3D**

Unity3D is a cross-platform game engine software which allows developing 2-dimension and 3-dimension video game. Not only some basic function such as light simulation, gravity simulation these kinds of physical support but also virtual reality and augmented reality games are able to be developed on it. The main programming languages are C and java script and can be programmed by its build-in software, Visual Studio which can be installed accompanying with Unity3D. Version of it is of little impact in our project and it is also convenient to transform the develop version of project into another one. At last we could build the project on more than 25 platforms and this time I choose the android platform which requires me to install Android build support. The installation process is easy to go through and we should only check the boxes ticked to confirm the required build-in software.

## **3.2 Vuforia Library**

Vuforia library is the package we should import in the unity if we want to develop augmented reality. Itself isn't a develop tool but requires a development platform where its capabilities can be explored. To store the recognized target marker into Unity, developers should visit its official website (<https://developer.vuforia.com/>). First of all, developers need to login (register if not obtain an account)and get a developer license key below the License Manager window. With this license key we could activate our augmented reality function in unity. Near the License Manager is the Target Manager where developers can store their database there and upload the target marker. Vuforia supports four kinds of target types. The first is single image target and after setting its size, name and raw image, Vuforia will evaluate it based on its feature number, the

more features a target have the more easily it can be detected and the higher ranking vuforia will give. The second target type is cuboid which contains six faces and we should assign six equal-Sized image targets for it. The third acceptable target type is cylinder and it contains top, side, bottom, three faces. The last one is 3D object and for this kind of target type we should use Vuforia Object Scanner Application to scan the object into file format. After adding all the targets we need in Vuforia's website, we could then download the database and select the right development platform.

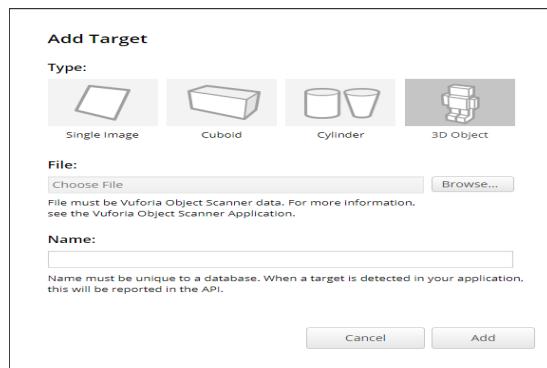


Figure 3.1: User interface of adding target in Vuforia

### 3.2.1 What is AR

AR's full name augmented reality and is an interactive experience of a real-world environment where the objects that reside in the real world are enhanced by computer-generated perceptual information. AR can be defined as a system that fulfills three basic features: a combination of real and virtual worlds, real-time interaction, and accurate 3D registration of virtual and real objects.[Wik20a]. The most AR games need to call the camera of device and put the unreal objects into the camera screen to reach a effect that real world seems to be augmented by unreal object.

For those people who aren't familiar with Informatics may get confused it with VR whose full name is virtual reality. The main difference between them is that VR implies a complete immersion experience and shut out the physical world. However AR relies on the physical world but just appends some other unreal object on it.

### 3.2.2 Vuforia Set-up

At the initial stage by using vuforia in Unity Editor we need to import Vuforia library into unity at the beginning. By clicking windows -> package manager a windows will pop up and we can search the Vuforia Engine AR in the left all packages list. After that, as mentioned above, we need to activate our vuforia function using the application license. key which was created in the previous subsection. Again, we click the window tab located at the above horizontal menu and find the Vuforia Configuration (There exists the shortcut key with **ctrl+shift+v** ). Right in the inspector view we could paste the license key there. And drag the slider bar down at the databases block, we could add the database by clicking the button and jumping to the vuforia website. After a series of preparation above it is done by the Vuforia set-up.

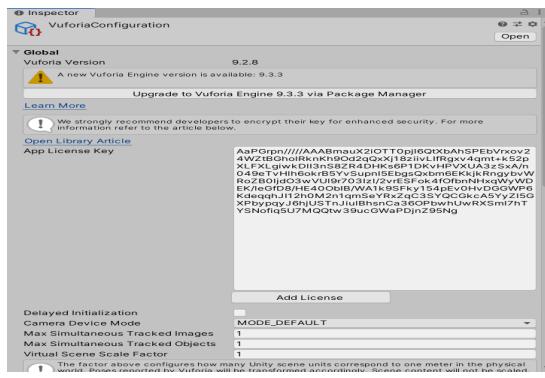


Figure 3.2: Vuforia configuration in the inspector view

### 3.2.3 Simplest AR example

One of the simplest AR cases is recognizing an image in the camera with a marker and a 3D model shows up above the marker or another image replaces the origin one in the screen. The implementation method isn't too much complicated and first of all we should create an AR camera in the scene. If we right click in the hierarchy view, a new option named "Vuforia Engine" is able to be created since Vuforia Library has already been imported.

The same with creating an AR camera, we could also find "Image Target" in the "Vuforia Engine" extended option list. This Image Target object obtains three default scripts, the first one is called Image Target Behaviour and defines which Image target or in other words image marker the object recognizes. And three types are available supported by vuforia, we could define the marker from database or pass the control right to the user or define with the help of a cloud Recognition. The last case is often used when

developer update the target frequently.

The created image target itself won't be seen in the camera but will be recognized when it appears in the sight of camera. And what is supposed to show up when the marker is tracked is the next step. It is realized by the second default script "DefaultTrackableEventHandler" and the function named "OnTrackingfound" is in figure 3.3 below.

```

    protected virtual void OnTrackingFound()
    {
        if (mTrackableBehaviour)
        {
            var rendererComponents = mTrackableBehaviour.GetComponentsInChildren(true);
            var colliderComponents = mTrackableBehaviour.GetComponentsInChildren(true);
            var canvasComponents = mTrackableBehaviour.GetComponentsInChildren<Canvas>(true);

            // Enable rendering:
            foreach (var component in rendererComponents)
                component.enabled = true;

            // Enable colliders:
            foreach (var component in colliderComponents)
                component.enabled = true;

            // Enable canvas':
            foreach (var component in canvasComponents)
                component.enabled = true;
        }

        if (OnTargetFound != null)
            OnTargetFound.Invoke();
    }
}

```

Figure 3.3: OnTrackingfound in DefaultTrackableEventHandler

The above code implements what should be happened when a target is tracked and found. The first three lines of code in "if" block get all the children's collider, renderer and canvas and enable them in the following three while blocks. That is to say the object which we expect to see when the target is found should be the child of the target in the scene. So, after creating it as the sub-object by the target and setting its position and scale properly, the simplest AR example is done.

### 3.2.4 OnTrackingfound and OnTrackinglost

These two functions are in the DefaultTrackableEventHandler that was talked about in the previous paragraph. But for accomplishing other effects not only appearance of the children these two functions can help developers in an easy way as well. Let's look at the last 'if block' in figure 3.3, OnTargetFound is a publish unityevent and defined in the beginning of the script, which isn't included in the screen cut in figure 3.3. An

unityevent is a zero argument persistent callback that can be saved with the Scene. This is the definition by the official unity website and clearly speaking, unityevent is more like a subscription event that can be edited in the inspector view. And the subscription event is a delegate which means a container of functions, if we add function in the container it is the same as we subscript something in the subscription list. And the last line of code in figure 3.3 is OnTargetFound.Invoke(), essentially it is unityevent.Invoke() and the function effect is invoking all registered callback-functions. So if we wanna add a sound effect when the target is found, easily could we just add the audio-play function in the list followed by OnTargetFound in the inspector view. If the reader have some unity experience, he could find the same point between this part and the button behaviour. In the inspector view of an UI button there exists a ButtonBehaviour script that uses the same unityevent way too.

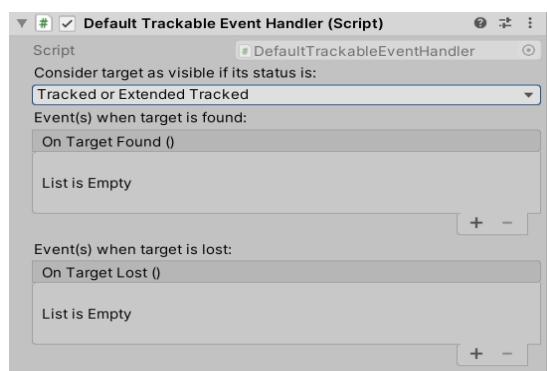


Figure 3.4: OnTrackingfound and OnTrackinglost method in inspector view

### 3.3 Jupyter Notebook

Jupyter Notebook provides with an online programming environment for Python language but indifferent with .py and .py3 files, it creates and accepts Jupyter notebook documents whose suffix extension is usually 'ipynb'. The reason of statement with 'usually' is because Jupyter can connect to many kernels to allow programming in many languages which currently includes some common programming languages such as java and Haskell. The installation ways are of many possibilities and I installed it by using anaconda which is convenient to install TensorFlow as well.

## 3.4 TensorFlow

TensorFlow is an end-to-end open source machine learning platform where users can create their models and train neural networks. It was originally developed by researchers and engineers working on the Google Brain team within Google's Machine Intelligence Research organization to conduct machine learning and deep neural networks research. The system is general enough to be applicable in a wide variety of other domains, as well. For using TensorFlow in Jupyter, an import of it is indispensable. And because of updates over years, developers should pay attention to the function which is out of date, that caused many problems when I programmed.

### 3.4.1 Definition of Deep learning

Before explaining the definition of Deep learning, it is essential to introduce what is machine learning. Machine learning(ML) is a computer algorithms which improves agents behaviour to be more rational by previous experience and data. It is a subset of artificial intelligent and is also a multi-domain crossover study which covers probability theory, statistics, algorithm complexity and so on. Traditional machine learning has several main research directions like Random forest, decision tree, Bayesian networks and artificial neural networks.

So Deep learning belongs to a part of machine learning and is based on feature representation learning with artificial neural networks which lets neural network learn by itself how to capture features. The inspiration came from the Noble prize owners David H. Hubel and Torsten Wiesel in 1959 with the theory of the primary visual cortex of the brain. They stated that there are two kinds of cells there, simple and complex cells, they take different responsibilities of the visual cortex function. And what's more, in the primary visual cortex layers the output of the first layer of complex cells serves as the input to the second simple cell layer and the rest can be done in the same manner.[Gra01] An example convolutional neural network[WZ19] is as shown in figure 3.5 below. Basically a neuron network contains an input layer, an output layer and several hidden layers, here in the figure obviously the input is an image, the output is firing rates and hidden layers are the layers in the middle. In both convolution layers and All-to-all layer there exist an activation function which has 'On' for 1 and 'Off' for 0 these two values. Activation functions act the similar behaviour as simple cell and the other layers such as All-to-all layer are a variant of complex cell.

Additionally, there are three types of data sets used in different purpose in deep learning. They are called training set, testing set and validation set. All the three sets store the same type of data which is able to convert to the required input shape in model. Training set is used to create the original model while validation set responds

to validate and improve the accuracy of the model and adjust the parameter. Normally the proportion between these two sets is about seven to three but depends on the way used to validate. Testing set is prepared for the last optimal model and evaluating, so it won't change the model and its parameters.

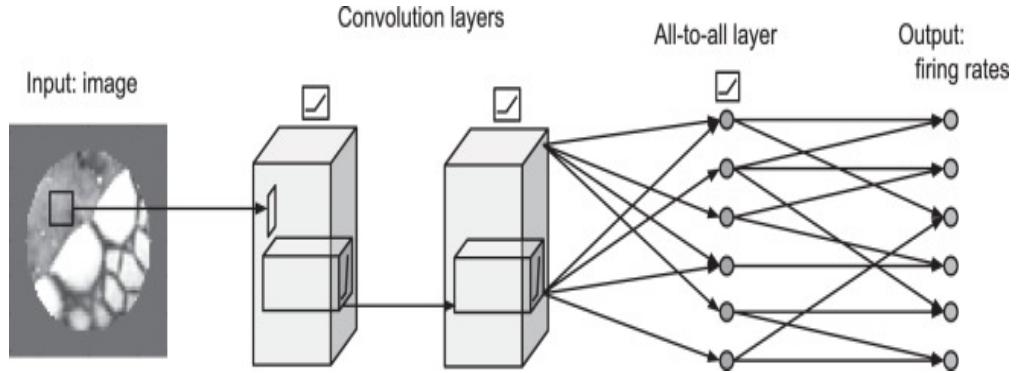


Figure 3.5: A common convolutional neural network (from [WZ19])

### 3.4.2 Google Colab

Google Colaboratory, or Google Colab for short is a research project by Google aiming at helping with transform learning. It uses Jupyter notebook environment working completely in the cloud and the frame such as Keras, TensorFlow, PyTorch and OpenCV are all available. No matter you are a student, data scientist or an AI researcher, this application is totally free for everybody while it also offers a Tesla K80 GPU which costs 1700 dollar in amazon. Since it is stored in the google cloud hard disk and can be shared just like Google document and Google Spreadsheets.

To train the model with a bulk of parameters and layers or a training set with thousand or even ten thousand data, it actually costs much time running the training progress. For my computer which is without GPU but only CPU, I spend about two minutes running one epoch when I trained the pretrained model like VGG16. But by comparison when I using Google Colab its free cloud GPU, the training time of each epoch only takes a few seconds.

## 4 Approach

### 4.1 Image comparison using TensorFlow

Apart from recognizing Japanese kanji when reading, writing kanji always has a place in Japanese learning progress. In Japanese writing system there are mainly two things which should be take good care of, One is whether a kanji is written correctly and the other one is the kanji stroke order which is a study requirement for Japanese local speakers. However, the latter is an over-high demand for foreigner learners and even in serious game so in this case this requirement can be ignored.

In the video game "kawaiinhongo" [Unk20b] there is a mini game asking player to write the kanji or hiragana according to the given correlated clew. After writing the kanji or hiragana, player needs to check in person if the writing is right or wrong.

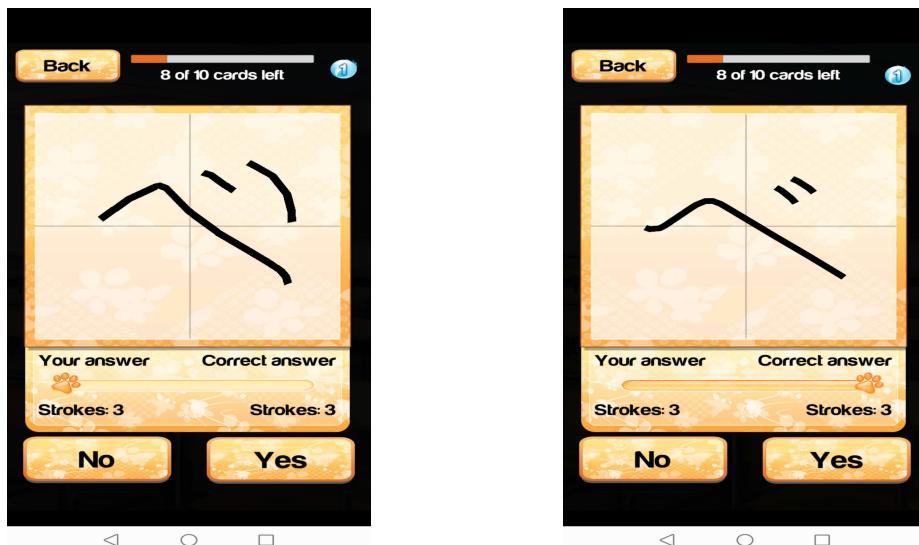


Figure 4.1: The hand-writing kanji in Figure 4.2: The correct hiragana in "kawaiinhongo"

So what if the game can recognize with some algorithms or techniques but not let

player do it personally? In this case I use deep learning technique with TensorFlow to build a model and train it so that it is possible to accomplish the function in game.

#### 4.1.1 Deep learning model

My deep learning model is created on Jupyter notebook and programmed in python. Limited by the performance of my computer's CPU and the size of data sets I plan to create a model which could only just distinguish three Japanese kanjis 朝(morning), 日(sun), 月(moon) in image and classify them. So the first step of creating the model is transforming the image data in training set and validation set to the pixel data. Here I imported CV, a library of programming functions aimed at real-time computer vision with its full name 'OpenCV'. Take into consideration of the non-uniformed size of images in data set, I used the following resize function and imread function transforming every input image into gray scale. According to the path of each image, clearly speaking in which folder the image locates, a corresponding class number will be given to it. All the image data including their category and pixel data is stored as a list in a list of list and can be used in the next training step.

I created my model as follows. In the first line I defined my model as a sequential model which means a linear stack of multiple network layers and is the simplest model in TensorFlow . Cause all the model needs shape of input data, the first layer of sequential model which here is the first convolution layer has to accept a parameter of input shape and in the following layers the input and output shapes will be automatically calculated and adjust.

Convolution is a function whose output is a outcome of multiple inputs. The aim of convolution layer is frequently image processing, such as smoothing, sharpening, and edge detection of images. The parameter name is the name of this layer, parameter filters equal to 16 means the dimensionality of the output space and kernel size(3,3) is height and width of window.

Normally there exist two common activation function, the one is called ReLu (Rectified Linear Unit) and the other is Sigmoid function. The difference between them is that Sigmoid function covers domain from 0 to 1 while ReLu covers a domain from 0 to positive infinity, that means Sigmoid function is usually used to model the probability and ReLu function is used to model all the positive number. Here I choose ReLu because it is more efficient when using CNN(Convolutional neural network).[Fer15] Max-Pooling layer takes the maximum value over the window defined by pool-size for each dimension and shift the window defined by strides in each dimension. The purpose of this operation is reduce the input size for the next layer and of course reduce the computing time consumption.

The output of convolution layers isn't able to fully connect with Dense layer so flatten layer offers the operation that aims at the multiple dimension output of convolution layers and transforms it to one-dimension.

Dense layer is also called fully connected Layer. Fully connected means that each neural in a layer receives an input from all the neurons present in the previous layer. The function of this layer is classifying the category and easily speaking is that every neural is on behalf of a feature holding a certain weight which decides the raw input belonging to which category.

After adding layers into model, `model.compile` is a function setting configuration for training and `model.fit` can train this model with parameter determine how and how long the model will be trained.

```
model = Sequential()

model.add(Conv2D(name="conv1",filters = 16,kernel_size = (3, 3),input_shape = x.shape[1:]))
model.add(Activation(activation = "relu"))
model.add(MaxPooling2D(name="maxpool1",pool_size = (2, 2),strides = (2, 2)))

model.add(Conv2D(name="conv2",filters = 32,kernel_size = (3, 3)))
model.add(Activation(activation = "relu"))
model.add(MaxPooling2D(name="maxpool2",pool_size = (2, 2),strides = (2, 2)))

model.add(Conv2D(name="conv3",filters = 64,kernel_size = (3, 3)))
model.add(Activation(activation = "relu"))
model.add(MaxPooling2D(name="maxpool3",pool_size = (2, 2),strides = (2, 2)))

model.add(Flatten())
model.add(Dense(512))
model.add(Activation(activation = "relu"))
model.add(Dense(3,name="outputlayer"))
model.add(Activation("softmax"))

model.summary()

model.compile(optimizer='adam',
              loss='sparse_categorical_crossentropy',
              metrics=['accuracy'])

model.fit(x,y,batch_size=3,epochs=25,validation_split=0.2)
```

Figure 4.3: How to build a model step by step

#### 4.1.2 Overfitting and underfitting

Overfitting and underfitting are also called overtraining and undertraining in machine learning. They are two problems on model resulting on bad prediction. Overfitting takes place when a model is not generalised. The result causing that may be a complex algorithm to build the model or in the term of deep learning a complicated layer structure of a model. Comparably, a small data set can also lead to this problem in the

other hand. To explain this more simply, overfitting is just like a model remembering all the data in the data set and what's bad is that even a single data can have influence on it. This isn't reliable as the word's general appearance, because the model can't predict well with the untrained test data. We always want to find the trend, not fit the line to all the data points,[Al-19].

Underfitting is on the contrary the problem indicating that a simple model can't fit the training data set with relatively big size. As follows is a figure explaining the two problems and in it the blue line indicates underfitting ,red line indicates overfitting and green line is a normal fitting.

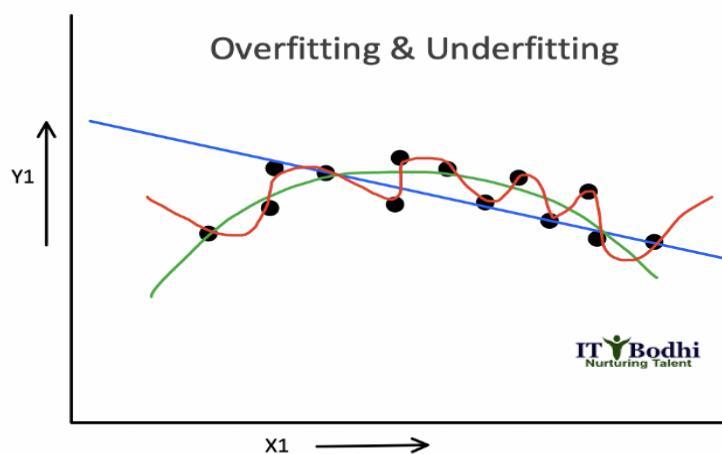


Figure 4.4: An example of underfitting and overfitting

#### 4.1.3 Data augmentation

Since all the data images in the training data set are drew by me, so owing to the confined number of samples the validation accuracy always keeps at a bad value. In the normal cases deep learning needs thousands of data to train a model well and dozens of images by my situation can't achieve that. To increase my data set, I used data augmentation technique so that a single image can be augmented to dozens of different data.

Data augmentation is a technique used to increase the amount of data by adding slightly modified copies of already existing data or newly created synthetic data from

existing data. Under the circumstances that image as a sort of data some image processing techniques such as resizing, rescaling or zoom are used. However, not all the image processing technique are allowed to use kanji comparison for example the mirror rotation and the normal rotation with an overlarge angle.

#### **4.1.4 Freeze the model**

When finish training the model, there comes a next step named freezing the model. Freezing is the process to identify and save all of required things(graph, weights etc) in a single file that you can easily use, and further more this function helps getting rid of unnecessary meta-data, gradients and training variables and encapsulate them all in a single file[Unk20a]. Because under normal circumstances neural networks are computationally very expensive and the number of variables required for prediction is very huge so that they are indeed a heavy load on a web server, which developer can use `model.summary()` to check the parameter details of each layer.

#### **4.1.5 Encountered problem**

As is mentioned in the previous subsection, TensorFlow has been updated from version 1 to version 2, more exactly is 2.3.0. So most of the functions are obsolete including the freeze function given in the forum stack overflow[Sol17]. It was very weird when I followed every steps with YouTube tutorial but still got various of errors which didn't seem to be solved easily. But luckily I found a adaptable function with TensorFlow 2.0 in the following website. [lei14]

#### **4.1.6 Prediction result**

The test images used for prediction are brand new to training phase, that means neither the original test images nor their augmentation are known by validation or training phase. After 50 epochs training my model, prediction result eventually turns to 100 percents accuracy. Below is a screen shoot showing the prediction result.

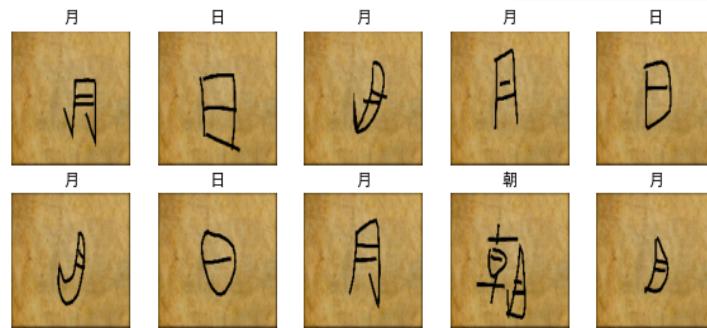


Figure 4.5: Prediction result

## 4.2 Load game data using XML file

### 4.2.1 General structure of XML

Extensible Markup Language (XML) is a markup language that defines a set of rules for encoding documents in a format that is both human-readable and machine-readable(by Wikipedia). This kind of file is pro to be used by storing list of game contents for example a great large list of items including many iterations on it. likewise, XML allows not only reading data but also writing and storing, so the created content in game by player can also be stored in it. But additionally, XML isn't ideal to save the local game content for instance the character data or game process. The reason is that the structure of XML is easy to understand by players and they could make changes which causes a frustrating game flow.

XML has a tree structure which is bound to possess a root and from it expands to leaves.(Take a look at Figure 4.6)

The first line of code is a processing instruction which begins with <? and ends up with ?>. The first word behind <? is XML statement and in this case it obtains three characters: version 1.0 stands for this file accords with 1.0 norm, standalone states if the document is in this file or needs to be imported exteriorly and encoding is the document character encoding. Starting with the root node and ends up with the end mark of root node, this document stores two penguins and each has three child nodes and one character. Every node is composed of begin mark, end mark and content. And if a node locates between the begin mark and end mark of an another node, it is on behalf of the parent-child relation.

```
<?xml version="1.0" encoding="utf-8" standalone="yes"?>
<the_South_Pole>
  <Penguin category="Jumpenguin">
    <Name language="en">Peter</Name>
    <Sex>male</Sex>
    <Age>2</Age>
  </Penguin>
  <Penguin category="Rapenguin">
    <Name language="en">Lili</Name>
    <Sex>female</Sex>
    <Age>3</Age>
  </Penguin>
</the_South_Pole>
```

Figure 4.6: A brief example of XML code

#### 4.2.2 code of XML part

In my project XML file is used as a random name data bank where prestores all the possible human name classified by gender. And I also use XML file for storing some kanji information such as its English meaning its hiragana spelling and description associated with Heisig method when the corresponding kanji marker is tracked and the kanji AR is shown in the form of animation.

By developing project using XML, in the first place, the head file must be imported at the beginning of the script. Here should be the "using System.Xml;". I achieved the data loading function from XML file by using the following references and programming functions:

1. XmlDocument Doc = new XmlDocument(); To declare a variable named Doc with XmlDocument type.
- 2.Doc.load(string path); To load a XML file to this newly declared XmlDocument by using the path where the XML file locates.
- 3.XmlNode Node = Doc.SelectSingleNode(NodePath); Doc is the XmlDocument which I have already declared and with SelectSingleNode(string NodePath) I could find any specified node in XmlDocument.
- 4.XmlNode.FirstChild. This reference will return the first child of a XmlNode.
- 5.XmlNode.NextSibling. This reference will return the next sibling of a XmlNode and a sibling relationship exists when two elements have the same parent.
- 6.XmlNode.InnerText. This reference will visit the content of the XmlNode and return a string type.

## 4.3 Apply deep learning model in Unity using TensorFlowSharp

### 4.3.1 TensorFlowSharp

TensorFlowSharp provides APIs for use in .NET programs, including C and F. These APIs are particularly well-suited to loading models created in Python and executing them within a .NET application. To get started in Unity and program in visual studio, I installed the SciSharp.TensorFlow.Redist package in NuGet package manager in visual studio.

[Unk 4]

### 4.3.2 Code of TensorFlowSharp

The code of TensorFlowSharp makes it possible to predict result by using a outer model created by TensorFlow in visual studio. As usual, the namespace named TensorFlow should be imported so that all functions related to TensorFlow are available.

```
public void Predict(string filename)
{
    using (var graph = new TFGraph())
    {
        graph.Import(File.ReadAllBytes("Assets/Resources/TensorFlow/saved_model.bytes"));
        using (var session = new TFSession(graph))
        {
            var runner = session.GetRunner();
            var tensor = ImageToTensorGrayScale(filename);
            runner.AddInput(graph["x"][0], tensor);
            runner.Fetch(graph["Identity"][0]);

            var output = runner.Run();
            var vecResults = output[0].GetValue();
            float[,] results = (float[,])vecResults;

            int[] quantized = Quantized(results);
            for (int i = 0; i < quantized.Length; i++)
            {
                Debug.Log(i + " : " + quantized[i]);
                if (quantized[i] == 1)
                    Debug.Log(Labellist[i]);
            }
        }
    }
}
```

Figure 4.7: Prediction function using pretrained TensorFlow model

TensorFlow can be regarded as a data structure named graph consisting of edges and points, where edges are tf.Tensor and points are tf.Operation. And since these graphs are data structures, they can be saved, run, and restored all without the original Python code. TFGraph is a class in TensorFlowSharp representing this kind of data structure and hence I could import my pretrained model in it using TFGraph0.Import. The

file format saved by TensorFlow in python is .pb which stands for protobuf however Unity can't recognize this kind of format so I need to convert to .bytes format. In the next step I initialize a runner and a tensor and use runner.addInput to log input data. In the set of parameter of this function, the first parameter is a string standing for the incoming port which here is the input layer of the model and its name is "x".ImageToTensorGrayScale is a self defined function that accepts an input file, change it to gray scale, resize it and stores pixel information to a matrix which can be convert to tf.Tensor at last. runner.Fetch returns index-th output of an operation and here is the output layer named "Identity".

The final output of this prediction is a vector whose length is equal to the number of categories. Then I use a self defined function named "Quantized" to make sure which category does the input image belong to.

#### 4.4 Painting function in Unity using LineRenderer

In my project there is a scene that asks player to write kanji in a painting box and this hand written kanji will be compared with formal Japanese kanji to judged whether player has a good command of this kanji or not. To accomplish the painting function in Unity it is supposed to introduce LineRenderer component.

LineRenderer is a unity component allowing user to draw lines with multiple segments. The line will be created based on the settings player set: 1. Line positions. This component includes the positions of start point and end point. 2. Thickness. This component determine how wide the line is. 3. Material. Different materials on the line may look differently. After setting these parameters, lines are successfully created and because computer can only record discrete things so basically a curve line consists of multiple straight lines.[sch16]

The rest work is defining the rule that clicking and dragging the mouse will create line. The solution to rest work relates to two Boolean parameters in input class. The One is input.GetMouseButtonDown and the other is input.GetMouseButton. The previous one is true in the frame when the mouse is pressed down and the latter one is true when the mouse is still held down. The principle is clear and when the mouse is pressed, create a gameobject with LineRenderer and when we move the mouse so will another line be added in the LineRenderer. When the mouse is released and pressed down again, repeat the same work like before.

## 4.5 Animation Design using extended Heisig Method

When kanji is tracked by camera, I think, not a fancy model is supposed to be exhibition but an animation or other forms of methods with which the serious game could release the high difficulty of learning foreign language by players. And as described in previous chapter, Heisig method is an excellent way helping remember Japanese kanji, so in this project I will follow the same way providing with a Heisig way for each trackable kanji. The picture below is a good example with Heisig method showing up in the Independence game "Learn Japanese To Survive! Hiragana Battle". The picture offers a explanation of the Kanji 火(Fire). [Flo18] It is worth learning and following that the game designers put the original kanji and the pictographic explanation together so that establish a clear comparison for players. Additionally, designers color the kanji red depending on the feeling about the fire and thus make a connection between the real fire and the kanji fire. As followed I will introduce the extended Heisig method existing in my project but owing to lack of recourse I can use the self-hand-painting in the form of lantern slide instead of video animation.

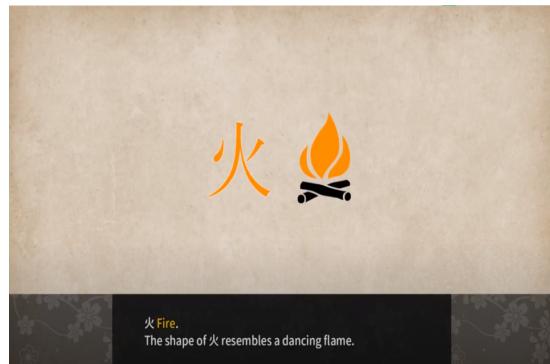


Figure 4.8: A shortcut in Learn Japanese To Survive! Hiragana Battle (from [Flo18])

### 4.5.1 instruction book method

The instruction book method is named by me and it was inspired from the example in figure 4.8. Despite painting and coloring on the original kanji to the real object, I comment by text description of each part which is like a instruction book guiding how to use or how to assemble a set of Lego building blocks. The benefit of it comparing to the example in figure 4.8 is that kanji doesn't have to be painted to a concrete real object. More clearly, the pictographic photo can be abstract but under the help of comments

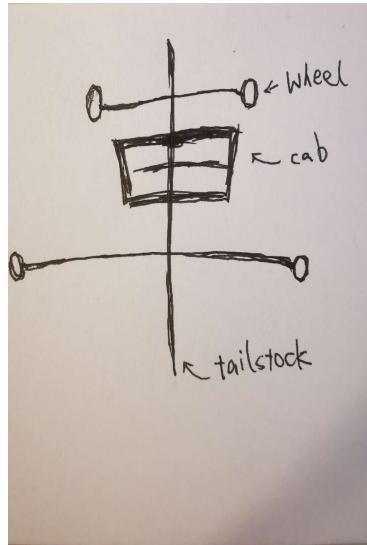
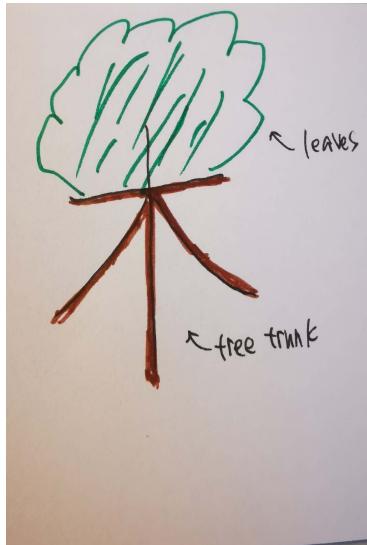


Figure 4.9: Instruction book method for 木 Figure 4.10: Instruction book method for 車

players or learners can image independently and think about whether the comments is right and how the pictographic kanji evolved from. This thinking process is helpful for stark the memory.

In figure 4.9 and 4.10 there are two examples of instruction book method, the one is a colorful kanji 木(tree) and the other is 車(car). Tree has a sense of green and brown so I color it for helping player remember but car obtains no fix color therefore is no need to do so. However both pictographic explanations are added rest part comparing to the standard Japanese Kanji, for example in figure 4.10 I add four wheels on the 車. So in order to not making confused between the standard one and explanation, the standard kanji should show up at the end of the animation.

#### 4.5.2 story method

Story method is a method telling player a small slice of story that helps them remember the kanji. The example below is a story about kanji "一"(One). It is a story about an ancient flying dragon knocked down a straight statue whose shape is exactly the roman numeral "one". Considering "一" is like a horizontally roman numeral one, in figure 4.13 the statue lying down on the ground . so making a connection between them will help player strengthen the memory. The kanji story below is also related to our game background which also intensifies the immersiveness. On the other hand, as is argued

in the previous chapter, humour is an effective help measure when it is attached in education. In the story of kanji one, the content or details can be designed humorously, which can be like a cartoon scene happening in "Tom and Jerry". More concrete for instance in this case, the dragon can feel easy to make the statue up in the beginning but actually takes a lot of effort.



Figure 4.11: Dragon find a statue

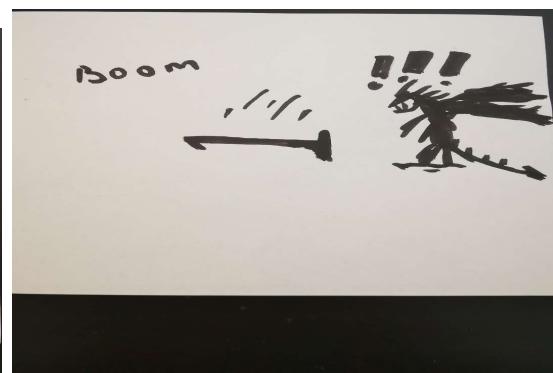


Figure 4.12: Dragon knocks the statue down



Figure 4.13: Dragon try to make it up

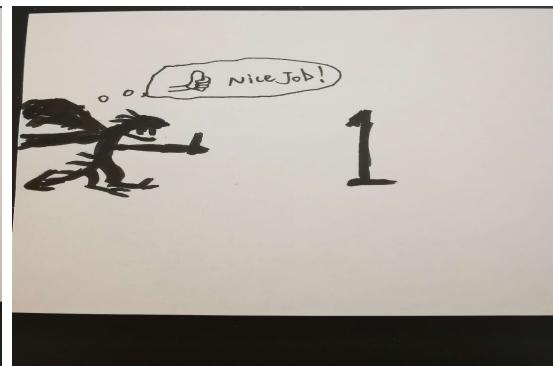


Figure 4.14: Finally the dragon did it

#### 4.5.3 Correlation method

Correlation method means putting the similar kanjis together and making an explanation by the correlated provenance. Simple Japanese kanjis are almost pictographic, so there exists correlation between them. The original Heisig method has already designed

#### *4 Approach*

---

explanation for the correlated kanjis such as (一)one and (二)two, obviously "二" can be regarded as two "一" stacked together and what's more "三" can be regarded as three "一" stacked together or just "一" plus "二". The example above can be the simplest and the most typical example which is always given in Japanese lessons. But in most cases there are some complex kanjis correlated with others, which is kanji with radical. This kind of kanji is basically correlated to its radical and obtains an additional meaning based on the other parts of kanji.

When game designer prepare to insert kanji with radical into the educational plan, I fiercely suggest to put the radical pattern aside from the complex kanji and restore radical's original meaning and appearance to strength the concept of radical.

# **5 Frame work**

## **5.1 Game Design Document**

### **5.1.1 Basic Information**

The game project is named as "Dragon Tale Prequel-The Origin" and aims to provide a new design thought for the developers working on the project "Dragon Tale – A Serious Game for learning Japanese Kanji ". The prequel game is totally made on unity and supports to run on the platforms of Android System and PC. The target customers are in the majority of the Japanese beginners at whatever age brackets whose first language doesn't belong to east Asian languages and they should preferably contains a certain amount of knowledge of Katakana and Hiragana as well. According to the value proposition canvas technique to analyse target customers, the pain customers suffering from should be taken into consideration. Since kanji isn't much familiar to those predefined customers, what my game should pay attention to offer is a reasonable way learning Japanese kanji and arousing their interest. So does my game project come into play in these aspects as is also the purpose of serious game. And to play this game, player needs two kinds of marker which are kanji marker and world marker. The usages will be mentioned in the following text.

### **5.1.2 Game Story**

At the beginning of all, world is dark and silent, the only thing can be heard is the breath sounds of the dragon god. Perhaps one day dragon can't bear the loneliness any more and it comes up with a idea to create other things to accompany with him. By using the ancient language(Japanese) the sun rises with light and warmness. Followed by the first human who knows nothing, dragon decides to teach him how to use number and furthermore to create the civilization.

The inspiration of this game comes from the other business simulation games. The story took place before the dragon tale as a prequel game. Player plays a role of dragon god and create the world from nothing. By teaching people Japanese kanji dragon god is able to bring civilization and world basic rules to the world. The Kanji is created by players themselves by means of combining the given clues to draw the kanji they

might think. Player may have a good sense of control as the world creator and find joy by finishing the creation and learn some common kanjis in a systematic educational approach during the game.

### 5.1.3 Level Design

Since the game project is only a prototype, only limited a few level can be designed. The prototype contains almost ten kanjis which are all common and can be frequently mentioned in daily Japanese conversations. The levels in game match with the evolution of civilization and nature. Take a brief example: The (日)Sun,(月)Moon and (人)Human are selected in the first level since the Sun is origin of the nature world and human is the base of civilization. And in the following level human begins to named each other and number is a indivisible part in the daily life, so name(名) and numbers such as one(一)two(二)three(三) are added in the second level. As levels go on, more kanjis should be unlocked in the same level and the complexity of selected kanjis increases as well.

## 5.2 Project Frame Work

The project consists of multiple scenes made by Unity3D and mini games serving for the purpose that is teaching the Japanese beginner kanji by a funny and effective approach. In this section all the mini games and Scene will be introduced.

### 5.2.1 Start Scene

The start scene of a game is just like face of a human, that means offering a first impression to the players. A game without a start scene misses integrity and can't guide the players' emotion into gameplay naturally and smoothly. Secondly, some basic settings needed to be set at the beginning should be on the start menu.

So my first scene of the project is a start menu demonstrating a demo world that in the later gameplay players could also achieve. And additionally four buttons are listed in the start menu which are play, settings, help and close at the top right corner. With the settings button player could reset the resolution and increase or decrease the main volume. To offer guidance for the green hand first playing this game, click the help button and then turn to the help panel, the text in the panel will explain how to play and what is the goal.



Figure 5.1: Start scene

### 5.2.2 Main Scene and Tech Tree

After clicking the play button in the main menu, the game comes to the main scene. The middle area in the main menu is supported for augmented camera and when first time in the main scene, a dragon on the left side which is exactly the role we play says that there exists nothing in the world and meanwhile nothing will appear if players show world marker. The tech tree button at the top left corner shows the kanji arranged by game levels, some words listed in the tech tree can be seemed as lessons or units in traditional Japanese causes. In the tech tree players could create objects on the world marker and various of interactive dialogues are attached to different kinds of game object.

For instance the Sun object stays in the sky all the time and doesn't even move in the beginning that the object is created. Players play the role as world creator so that any nature laws can be defined as they like and in this case players could change the size of the Sun or let the sun rise by day and set by night and even more a moon can be created as a "working partner" with the sun. However these changes aren't easily achieved without any effort. If players wanna create the moon, they should define the moon first as a precondition, more clearly they should know how to write the moon in the form of Japanese kanji as the goal to entice them learning kanji. Since (日) the Sun belongs to the first game level so taking the game design of difficulty gradients into consideration only simple and kanjis like 月(Moon)大(Big)小(Small)朝(Morning)午(Noon) are required to be learnt following the 日(Sun).

Another example can be the second words in unit one which is the Human(人). Obviously player can create a human with this word but when check the information of this human by clicking it on the world marker, it shows that the name, age and gender are the value of unknown. Hence, player should learn the kanjis about numbers for age ,the kanji name(名) for name and man(男),woman(女) to define the gender.

### 5.2.3 Marker Scene

In this scene player are asked to show the corresponding kanji marker with the given English meaning. when the meaning matches the required kanji marker, the animation using Heisig method will play and below in the dialogue box the relevant explanation in the form of text will be offered as well.

### 5.2.4 Quests panel

The button linked to the quests panel locates at the left top corner in main scene. The quest in quests panel, put in another way to say ,is the units in a teaching system. By translating the stiff educational requirements to joyful quests in the gameplay, the quests panel plays such a role in the project.

### 5.2.5 Drawing game

Some kanjis are learnt by showing the right marker in the marker scene but some other are not. To enrich the gameplay and the education form, a mini game asking player to draw the kanji using Heisig method with some given clues is planed as the main gameplay in my project. Based on the theory of Edgar Dale as argued in the previous chapter, not only a visual education form like showing the right marker, but also a more abstract and practical way like this counts for serving different kinds of customers.

A concrete example of the kanji (月) moon is from the sun unit because developer could find a proper Heisig method based on the similarity between the both. The animation of sun(日) shows that the sun can be split into two parts: the outline border comes from the shape of then sun but due to the diamond character of Japanese kanji, it turns to square replacing with circle and the horizontal line in the border stands for the sunlight.

Take the Heisig method using on sun(日) as reference substance, I select these characters from moon (月) and use metaphor as a kind of clue.

1. The moon(月) is brighter than the sun (means two horizontal lines in the middle)
2. The moment dragon saw the moon, it is curved at that time(means the hook and the slim curved shape)

## 5 Frame work

---

3. (Show an image that the left side of the moon is covered and exactly matches the shape of kanji moon (月)



Figure 5.2: Drawing game : Moon(月)

It works on the complicated kanjis as well, for an other instance the compound kanji morning(朝) has far more strokes than the previous mentioned kanjis but however still has a shortcut to be remembered with Heisig method. The clues are :

1. In the morning dragon could see the sun.(A sun(日) in the morning (朝))
2. The sun rises from the horizon(refers to the horizontal line on the left)
3. The sun movement direction is upwards(a upwards arrow is like the vertical line on the left)
4. In the early morning you could also see the moon on the right



Figure 5.3: Drawing game : Morning(朝)

### 5.2.6 Feedback

The pattern drawn by players may have great difference with standard Japanese kanji but is in all probability like the Kanji in the ancient time with its original pictographic meaning. After finishing the painting and clicking the confirm button the game will check the pattern right or not using deep learning and if the kanji is drawn in a right way, the standard kanji will show as a comparison and a sample solution to strength the learning memory. If the kanji is drawn wrong, it will plainly output "You wrote the kanji incorrectly, please try again". Afterwards the drawing area will be cleaned and let player try again.



Figure 5.4: Drawing game : Morning(朝)

### 5.3 Game Flow Analyse

Video games are often conceived as a series of puzzles that use rewards to motivate completion. Bleszinski describes this process as 'game flow'. [Unkwn] The sentences above are the original definition of game flow. And back to my project, the puzzle is drawing kanjis according to given clues and remembering kanjis with the help of Heisig method. When finishing the puzzle, players could add new element or define new rules in the world that is dominated and created by themselves. More directly, when using augmented reality, players could see the created world in the real environment by showing the world marker in front of the camera and this interaction can be tuned towards facilitating flow. In this part especially, players can get tremendous sense of achievement for writing out the unfamiliar kanjis which hasn't been seen before. The feeling doesn't only originate from overcoming difficulties but also the self-approval in the study of eastern language. For years the western people think that eastern culture

has a sense of mystery, but in the help of Heisig method the pictographic language takes off its mysterious veil and can be easily got clear.

# **6 Evaluation**

Evaluation is absolutely an necessary part in the progress of developing a commercial video game. But limited by the social circumstance now, an actual interview with a large number of participants is impossible. If the situation of corona 19 gets stable in some days, the evaluation plan should work then.

## **6.1 Evaluation before development**

An advance business evaluation of a project can avoid unnecessary financial losses, improve the product quality and fit the requirements of costumers more. To evaluate the product before starting development, the first step includes defining the target group and analysing whether the product has relatively great competitiveness among the market. If the answer is "not really", then developers should make changes on it by using brain storming method to find good ideas serving target group's interests.

### **6.1.1 Target group analyse and Personas**

The target group mainly aims at the Japanese beginners from non-Asian area with basic knowledge of katakana and hiragana. No matter how old they may be and just as well whether they are students or not this game is supported to everyone who wants to get access to Japanese. This type of target group however, in the most cases hasn't been exposed to kanji system before and the complicated kanjis may cause great difficulty for them to be remembered. So to solve this problem, a core gameplay surrounding Heisig method was considered to be developed in my project.

Personas is usually a graphics recording some information of a real or imaginary user. Some basic information such as name, age, gender, experience including social life and individual life and hobbies are in most cases listed in personas for research and improve product quality [DS20]. To evaluate my project, I got contacted with some Chinese and Italian classmates with whom I attended Japanese classes last year. Chinese classmate may not in the main target group of my project because Japanese kanjis are easy to Chinese, but owing to the lock down situation, half a loaf is better than no bread. I created personas of them after short interview finding them all like the simulation game, so I followed their likeness to create my project.

## 6.2 Expected results

Serious games always expect results with certain purpose. For my project which is unfortunately a demo limited by time and resources, the expected results are building self-confidence for player that is getting rid of the fear of learning kanjis, supporting an efficient method and concept to remember kanji and learning the kanjis appearing in my demo project.

Kanji learning is more than just pronunciation and meaning, how to write kanji is a task full of challenge. When writing kanji wrong, it may result in frustrating feeling and destroy learner's interest and self-confidence. The drawing game to guess how a kanji look like will bring great sense of achievement and helping build self-confidence. Heisig method is a well-known kanji learning mechanic which should be introduced to all the Japanese beginners. Furthermore, under the consideration of different kinds of learner and using Edgar Dale's theory, Heisig method got optimized in my project and could probably help in remembering kanjis.

The kanjis appearing in my project are basically all the numbers from one to ten and some common kanjis including sun(日),human(人) and so on. The game duration last about 15 minutes and is of course enough to remember them as an introduction to learning Japanese kanji.

## 6.3 Evaluation after development

To check whether the expected results can be achieved or not, the evaluation after development should be executed. The plan of it is firstly defining participants and sending questionnaires to them. When collecting all the feedback from participants, the last result of evaluation can come to a conclusion.

### 6.3.1 Participants

Participants of the second evaluation should be the more the better, because the more the number of sample is, the more stable and more reliable the result can be. All the participants should have intention learning whatever a foreign language which means "not limited to Japanese" and can have no knowledge of katakana and hiragana. The participants joining the evaluation is different from the target group mentioned above and should have a wider selecting range than target group. The benefit of this is to gather information from more people as possible and to test whether the project can still appeal to other people who wasn't going to learn Japanese.

Participants' basic information such as age, gender and nationality should be offered for evaluation. But name is no need to know so take into consideration of personal

privacy, the questionnaire can be anonymous. Moreover, it is inquired how long does the participant surf the internet and play video game including on PC and mobile phone. Japanese language level and the feeling about learning Japanese kanji also count for the last evaluation conclusion.

### 6.3.2 Questionnaire and System Usability Scale

The questionnaire I designed for my project consists of several basic questions and a complete System Usability Scale. System Usability Scale or SUS for short is a questionnaire to measure usability of a system. The SUS has a ten-items attitude Likert scale that every item can be marked within five scores(from fully agreement to fully disagreement). Questions in the SUS cover three kinds of aspect which are effectiveness, efficiency and satisfaction.

Effectiveness means the ability measuring how well does user use this project to finish some task and how good the last result is.

Efficiency means the level of resource consumed in performing tasks which it says in [Bro08]. Here it can be the time spent on learning kanjis by using this project.

Satisfaction is a positive subjective feeling towards this project.

The Classes of metric can vary for concrete analysis of concrete problems. They basically lies on the type of system and which kind of result system outputs. So the ten questions in my SUS will slightly deviate from how it defines.

### 6.3.3 Evaluation result from Questionnaire

The ultimate result will be analysed and determined by the SUS scores weighting with user's basic information. For instance the participants who is learning Japanese should have heavier weight than others, and elder people should have lighter weight than younger people, because younger people are in the target group of a normal commercial video game with their higher level of consumption.

SUS scores sums up the scores of ten items from the SUS. For the sake of not continuing using positive or negative statements to induce participants' subject behaviour and judgement, questions are shuffled and separated with the same tone. For example questions 1,3,5,7 and 9 will get more scores when the scale is checked in a higher position. And on the contrary, other questions will get more scores when the scale is checked in a lower position. So according to this rule, the SUS scores are equal to  $p-1$  in the place of 1,3,5,7,9 and equal to  $5-p$  in the place of 2,4,6,8,0 where  $p$  interprets the position. And at last the SUS scores will multiply by 4 since the most score one question will get is four and this calculation will let the full score reach one hundred. [Kla]

## Dragon Tale Prequel - The Origin

### Questionnaire

1. How old are you?

2. Are you a man or woman?

- man  
 woman

3. What's your nationality?

4. How long do you play video game every week?

5. Do you know a few Japanese kanji?

Figure 6.1: Questionnaire

6. Do you learn Japanese now?

- yes
- no

7. System usability scale

	Totally wrong	not really	general	right	Totally right
I had fun playing this game	<input type="radio"/>				
I felt the game difficult	<input type="radio"/>				
This game inspired me in the way of remembering kanji	<input type="radio"/>				
I still think it hard to learn Japanese kanji	<input type="radio"/>				
I can now remember how to write sun, moon and morning in Japanese	<input type="radio"/>				
The game can't appeal me continually playing	<input type="radio"/>				
I am satisfied with this project	<input type="radio"/>				
I think the game play need to optimize and improve	<input type="radio"/>				
I will buy this game once the official edition is released	<input type="radio"/>				
This game can't solve my problem on learning Japanese	<input type="radio"/>				

Figure 6.2: Questionnaire

# **7 Future work**

Until now this project still needs some improvement on the way to be sold on market. The project is basically a demo now whose gameplay can only last 15min. The aim of this project is just as expected results in Evaluation chapter, to test whether Heisig method helps in that form learning Japanese kanji for beginner.

## **7.1 Gameplay improvement**

Gameplay acts a key part in serious game attracting player to keep on playing. Actually the demo game more focused on educational design rather than gameplay design. In summary the gameplay in demo is composed of drawing game, marker showing game and additionally the gameplay creating and designing a virtual world. To be honest, although these gameplay can bring fun to player but it is still far from being desired. So in the following space of this section I will discuss about the improvement separately from various of aspects.

### **7.1.1 Game theme and game genre determining**

Game theme refers to the subject matter the game is built around.[20a] The so-called subject matter can come from history like cold war or Egypt , or from the imaginary such as zombie or Cyberpunk. Around these subject matters the storyline can be designed which brings the game a sense of immersion. However, it is absolutely remarkable that game theme isn't equal to the definition of game genres. Game genres are usually used to describe the type of video game and this so-called type is classified by gameplay. [20b] For a brief example, A game with zombie theme can be either a FPS(first-person shooters) game or strategy game, among which FPS and strategy belong to game genres.

So what kinds of theme and genre does my project have with a considerable development prospect? As far as I concerned, the solution from this question may lie on how casual or how hardcore the serious game is designed to be. If the future project tends to casual, it is a good idea to use the theme about dragon combined with education simulation as game genre. Just like the baby dragon idea in "Dragon Tale", the project can follow the example of SIMS [Inc20], but then the simulating character isn't a single

person but a dragon. The storyline can unfold as the birth of baby dragon when creating the world and can be mainly around the progress of bringing up the dragon.

The second idea is more suitable when the game tends to more hardcore. To add some challenges in the game, the created world can be filled with unpredictable difficulties including but not limited to coldness or excess temperature. Inspired by frostpunk [stu20] that is a survival game building city and formulating laws and regulations in an ice world. As in the figure 7.1, the challenges of gameplay in frostpunk is meeting



Figure 7.1: Frostpunk screenshot from[stu20]

the needs of various resources and satisfying the citizens so that they don't fire you to leave the place of mayor. The similar principle can be utilized in my project. When creating the world, the forces of evil like pirates or a katakiyaku (villain) in dragon tale can be the challenge as well. To survive from the trace of evil or protect the world from being destroyed, the project can become either a strategy education simulation game or an adventure game.

### 7.1.2 Game storyline and game world view

Review the some successful video games published in the past we could find something in common related to their hot sale and popularity. World of Warcraft has a huge world view and even a complete history line throughout the game[Bli20b]. Starcraft contains multiple character races and there will always exist a few "famous" characters in each

race with their own background story[Bli20a]. Even the MOBA(Multiplayer Online Battle Arena) game like League of Legends want to tell players a background story despite the story won't influence on it[gam20]. Playing a game with rich and attractive story is just like reading a novel, which will easily polish the gameplay with literal charm.

With regard to the storyline in my project, in my opinion it can integrate Japanese culture and use some unique Japanese mystery to enrich the of Japanese theme. For example, Orient dragon, ninja and sakura can represent Japanese culture and the storyline can be made up with these elements.

## 7.2 Resource improvement

The resource here involves the animation, 3D model and training model for deep learning etc. If the project is considered to be put on market, finding a professional team or asking for outsourcing company is an optional way to solve the resource problem.

### 7.2.1 Animation and 3D model

While putting the marker front of AR camera, the animation video will appear to show how the kanji can be remembered using Heisig method. The animation here covers the origin marker and just plays a motion graph consisting of several hand drawing images. Hence, there are still some space of improvement in that aspect.

One improvement option is drawing more images so that the animation could look more fluid.

The second improvement option is using 3D model to produce animation instead of hand drawing images. To bring AR's superiority into full play, 3D model standing on the marker is more stereoscopic and real.

Besides the resource using for animation, the 3D model appearing in the scene should be replaced by others. These models come from Unity Assets Store and are all free. For commercial use, a finer set of 3D model resource is worth investing.

### 7.2.2 Training model

The training model of deep learning used to recognize the picture pattern is still unstable and a little bit unreliable. Although many techniques are used such as cross validation and data augmentation, the prediction can't guarantee accuracy close to one hundred percents for every kinds of prediction inputs. The most direct improvement method is increasing the data amount and the data is preferably all original images

not the images from data augmentation. The training and validation data using on my deep learning model for now is much insufficient and in the normal case the data number should be thousand times more than mine.

Additionally, the layers added to the model also have an effect on the accuracy of prediction. For example VGG16 model is a pretrained convolutional neural network model using for image detection and classification[18]. Even though this model used a data set of over 14 million images for training for weeks, only 1000 kinds of objects can be correctly classified. But as for those images with kanji, VGG16 can't ensure a good result as well. For the question which kind of layer structure fits kanji detection well doesn't have a sample solution and the adjustment of layers should be done time and time again until an approving structure shows up.

## 8 Conclusion

It is a pity that due to the lack of questionnaire feedback, the project isn't able to evaluated whether it actually achieve the goal. But in a comprehensive perspective on the success of learn Japanese to survive series game[ste], the ideas of mine may have a bright future and potential commercial value. In the process of developing my project, there were also some problems about deep learning. Especially the interface between C and Python, only little information can be found on the internet. So overcoming these problems can be a part of achievement in the project as well.

Finally, It's my great pleasure showing my own opinion of learning Japanese using on serious game. I hope that my special idea about extended Heisig method and the drawing game can help on Dragon Tale and be a part of it.

# List of Figures

2.1	How do serious games help remembering (from [Pea16]) . . . . .	3
2.2	Player encounters positive emotion in the game flow zone (from [Che07]) . . . . .	6
2.3	Player encounters negative emotion out of the game flow zone (from [Che07]) . . . . .	6
2.4	The expanded game flow zone (from [Che07]) . . . . .	7
2.5	Game flow adjustment through choices (from [Che07]) . . . . .	7
2.6	A screenshot of Koe(demo) from YouTube teaching complicated word (from [The17]) . . . . .	12
2.7	A screenshot of Koe(demo) from YouTube teaching easy expression (from [The17]) . . . . .	13
2.8	Dale's Cone of Experience (from [BAA13]) . . . . .	15
2.9	Multimedia Cone of Abstraction (from [BAA13]) . . . . .	15
2.10	How is friend explained in "Remembering the Kanji" (from [Hei07]) . . . . .	17
3.1	User interface of adding target in Vuforia . . . . .	20
3.2	Vuforia configuration in the inspector view . . . . .	21
3.3	OnTrackingfound in DefaultTrackableEventHandler . . . . .	22
3.4	OnTrackingfound and OnTrackinglost method in inspector view . . . . .	23
3.5	A common convolutional neural network (from [WZ19]) . . . . .	25
4.1	The hand-writing kanji in "kawaiinhongo" . . . . .	26
4.2	The correct hiragana in "kawaiinhongo" . . . . .	26
4.3	How to build a model step by step . . . . .	28
4.4	An example of underfitting and overfitting . . . . .	29
4.5	Prediction result . . . . .	31
4.6	A brief example of XML code . . . . .	32
4.7	Prediction function using pretrained TensorFlow model . . . . .	33
4.8	A shortcut in Learn Japanese To Survive! Hiragana Battle (from [Flo18]) . . . . .	35
4.9	Instruction book method for 木 . . . . .	36
4.10	Instruction book method for 車 . . . . .	36
4.11	Dragon find a statue . . . . .	37
4.12	Dragon knocks the statue down . . . . .	37
4.13	Dragon try to make it up . . . . .	37

*List of Figures*

---

4.14 Finally the dragon did it . . . . .	37
5.1 Start scene . . . . .	41
5.2 Drawing game : Moon(月) . . . . .	43
5.3 Drawing game : Morning(朝) . . . . .	43
5.4 Drawing game : Morning(朝) . . . . .	44
6.1 Questionnaire . . . . .	49
6.2 Questionnaire . . . . .	50
7.1 Frostpunk screenshot from[stu20] . . . . .	52

# **List of Tables**

bibliography

# Bibliography

- [18] *VGG16 – Convolutional Network for Classification and Detection.* website. [Online]. 20 November 2018.
- [20a] *Game Themes.* website. [Online]. 5-Nov-2020.
- [20b] *The Evolution of Video Game Genres.* artical. [Online]. January 26, 2020.
- [Al-19] A. Al-Masri. *What Are Overfitting and Underfitting in Machine Learning?* [Online]. 22-Jun-2019.
- [ali13] K. alive. *The 214 traditional kanji radicals and their variants.* [Online]. 2013.
- [BAA13] C. E. Baukal, F. B. Ausburn, and L. J. Ausburn. “A proposed multimedia cone of abstraction: Updating a classic instructional design theory.” In: *Journal of Educational Technology* 9.4 (2013), pp. 15–24.
- [Bee06] R. E. Beermann. *Introduction to the Japanese Writing System* 文字入門. [Online,accessed 14-Nov-2020]. 2006.
- [Bli20a] Blizzard. *starcraft.* official website. [Online]. 5-Nov-2020.
- [Bli20b] Blizzard. *World Of Warcraft.* official website. [Online]. 5-Nov-2020.
- [Bro08] J. Brooke. *SUS - A quick and dirty usability scale.* artical. [Online]. 30-Jul-2008.
- [CB10] J. Cannon-Bowers and C. Bowers. *Serious game design and development: technologies for training and learning: technologies for training and learning.* IGI global, 2010.
- [Che07] J. Chen. “Flow in games (and everything else).” In: *Communications of the ACM* 50.4 (2007), pp. 31–34.
- [Con11] T. Connolly. “Leading Issues in Games Based Learning.” In: Academic Conferences Limited. 2011.
- [DAJ11] D. Djaouti, J. Alvarez, and J.-P. Jessel. “Classifying serious games: the G/P/S model.” In: *Handbook of research on improving learning and motivation through educational games: Multidisciplinary approaches.* IGI Global, 2011, pp. 118–136.
- [Dic05] M. D. Dickey. “Engaging by design: How engagement strategies in popular computer and video games can inform instructional design.” In: *Educational technology research and development* 53.2 (2005), pp. 67–83.

## Bibliography

---

- [DS20] R. F. Dam and T. Y. Siang. *Personas – A Simple Introduction*. artical. [Online]. 7-2020.
- [Fer15] Ferdi. *What are the advantages of ReLU over sigmoid function in deep neural networks?* Forum. [Online]. 2015.
- [Flo18] Flotaku. *Learn Japanese to Survive (Kanji) - Let's Play [Deutsch / 02] - Die ersten Kanji (水、火、木、石、金)*. Youtube. [Online; accessed 09-September-2020]. 2018.
- [GAM] S. GAMES. Koe. official website. [Online,accessed 14-Nov-2020].
- [gam20] R. game. LOL. official website. [Online]. 5-Nov-2020.
- [Gie15] H. W. Giessen. “Serious games effects: an overview.” In: *Procedia-Social and Behavioral Sciences* 174 (2015), pp. 2240–2244.
- [Gra01] P. Grace W. Lindsay. *Convolutional Neural Networks as a Model of the Visual System: Past, Present, and Future*. artical. [Online]. 2001.
- [Gro19] P. D. G. Groh. *Social Gaming/Social Computing-Basics of Game Theory*. [Online,accessed 14-Nov-2020]. 2019.
- [Hei07] J. W. Heisig. *Remembering the kanji, volume 1: A complete course on how not to forget the meaning and writing of Japanese characters*. Vol. 1. University of Hawaii Press, 2007.
- [HLA13] M. J. Hays, H. C. Lane, and D. Auerbach. “Must Feedback Disrupt Presence in Serious Games?” In: *AIED Workshops*. Citeseer. 2013.
- [Inc20] 2. E. A. Inc. SIMS. official website. [Online]. 5-Nov-2020.
- [Jia07] S. Jiang. “Into the source and history of Chinese culture: Knowledge classification in ancient China.” In: *Libraries & the cultural record* (2007), pp. 1–20.
- [Juu10] J. Juul. *A casual revolution: Reinventing video games and their players*. MIT press, 2010.
- [Kla] P. D. phil. Klaus Bengler. *Evaluationsmethoden II*. artical. [Online].
- [KOI10] KOICHI. *THE HISTORY OF KANJI WHERE DID KANJI COME FROM, ANYWAY?* [Online]. MARCH 22, 2010.
- [lei14] leimao. *Frozen Graph TensorFlow*. GitHub. [Online]. 2020 Aug 14.
- [Lom12] I. Lombardi. “Not-so-serious games for language learning. Now with 99, 9% more humour on top.” In: *Procedia Computer Science* 15 (2012), pp. 148–158.
- [Pea16] A. Pears. *Why Serious Games? 6 Key Benefits*. [Online]. 5.08.2016.

## Bibliography

---

- [sch16] U. school. *Drawing Lines with LineRenderer*. [Online]. March 21, 2016.
- [Sol17] Solix. *How to export Keras .h5 to tensorflow .pb?* forum. [Online]. 2017.
- [ste] steam. *Learn Japanese To Survive! Hiragana Battle*. [Online; accessed 13-Nov-2020].
- [stu20] 1. studio. *frostpunk*. official website. [Online]. 5-Nov-2020.
- [The17] TheHeroOfLight. *Koe (Demo)*. Youtube. [Online; 2017].
- [Unk 4] Unknown. *TensorFlowSharp*. GitHub. [Online]. 2019 Dec 4.
- [Unk14] Unknown. *Why We're More Likely To Remember Content With Images And Video (Infographic)*. artical. [Online]. 9-18-14.
- [Unk20a] Unknown. *Freeze Tensorflow models and serve on web*. official website. [Online]. 2020.
- [Unk20b] Unknown. *Learn Japanese with Riko and her friends in kawaii*日本語! official website. [Online]. 2020.
- [Unkwn] Unknown. *What is Gameflow? Shrink and Grow*. [Online]. Unknown.
- [Wik20a] Wikipedia contributors. *Augmented reality — Wikipedia, The Free Encyclopedia*. [Online; accessed 22-August-2020]. 2020.
- [Wik20b] Wikipedia contributors. *Forgetting curve — Wikipedia, The Free Encyclopedia*. [Online; accessed 5-July-2020]. 2020.
- [Wik20c] Wikipedia contributors. *Maslow's hierarchy of needs — Wikipedia, The Free Encyclopedia*. [Online; accessed 7-July-2020]. 2020.
- [WZ19] E. D. C. William F. Kindel and J. Zylberberg. *Using deep learning to probe the neural code for images in primary visual cortex*. artical. [Online; accessed 2019 Apr 26.] 2019.